Gestion de la qualité de service (QoS) dans un réseau LoRaWAN avec mobilité

Norhane Benkahla

THÈSE DE DOCTORAT

PRÉSENTÉE À

l'ÉCOLE SUPÉRIEURE DES COMMUNICATIONS DE TUNIS

POUR OBTENIR LE TITRE DE

DOCTEUR

en Technologie de l'Information et de la Communication :

PAR

NORHANE BENKAHLA

SUP'COM
Higher School of Communication of Tunis

Gestion de la qualité de service (QoS) dans un réseau LoRaWAN avec mobilité

Devant le jury composé de :

Pr. Nabil TABBANE	**Président**
Pr. Tijani CHAHED	**Rapporteur**
Pr. Lamia CHAARI	**Rapporteur**
Pr. Chiheb REBAI	**Examinateur**
Pr. Mounir FRIKHA	**Directeur de thèse**
Pr. Ye Qiong SONG	**Co-Directeur de thèse**
Dr. Hajer TOUNSI	**Encadrant**

21-06-2021

Table des matières

Introduction générale **1**
Contexte général 1
Objectif et contributions 2
Structure du manuscrit 4

I Les technologies IoT & LoRaWAN **5**
1 Introduction 5
2 Les réseaux LPWANs 8
2.1 DASH7 8
2.2 WEIGHTLESS-SIG 9
2.3 Sigfox 10
2.4 NB-IoT 11
2.5 LoRa/LoRaWAN 11
2.6 Synthèse et choix technologique 12
3 LoRa/LoRaWAN 15
3.1 Modulation Chirp Spread Spectrum (CSS) dans LoRa 15
3.2 Caractérisation des paramètres de transmission LoRa 16
3.2.1 Facteur d'étalement (SF) 17
3.2.2 Bande passante (BW) 17
3.2.3 Taux de codage (CR) 17
3.2.4 Puissance de transmission (TP) 17
3.3 Impact de la variation des paramètres sur la performance du réseau . 18
3.4 LoRaWAN 19
3.4.1 L'architecture du réseau LoRaWAN 20
3.4.2 Classes des noeuds LoRa 20
3.4.2.1 Classe A (All) 21
3.4.2.2 Classe B (Beacon) 21
3.4.2.3 Classe C (Continious) 22
3.4.3 Limitation du Duty Cycle 22
3.4.4 Temps de transmission - Time On Air (ToA) 23
3.4.5 Procédures de jointure (ABP et OTAA) et Sécurité 24
3.4.5.1 Procédure OTAA [42] 24
3.4.5.2 Procédure ABP [42] 24
3.4.6 Format du message 24
3.4.6.1 Format du message MAC 25
3.4.6.2 Format des messages de jointure 26

3.4.7 Sécurité dans LoRaWAN 27
3.4.8 Procédure d'adaptation de débit (ADR) 29
4 Conclusion 32

II Extension d'ADR pour le support de la mobilité 33
1 Introduction 33
2 Les variantes ADR dans la littérature 36
2.1 Évaluation de performances de ADR dans un contexte de mobilité . . 36
2.2 Adaptation de la Configuration des transmissions descendantes 36
2.3 Choix des paramètres de configuration LoRa basé sur le sondage . . . 38
2.4 Contrôle adaptatif du débit orienté noeud basé sur un contrôle de congestion du réseau 39
2.5 Allocation adaptative équitable du débit de données et contrôle de la puissance dans LoRaWAN 40
2.6 Adaptation du débit en fonction de la qualité moyenne du signal . . . 41
2.7 Adaptation de débit en utilisant tous les SF 41
2.7.1 EXPLoRA-SF 41
2.7.2 EXPLoRA-AT 42
2.8 Approche méta-heuristique pour l'optimisation des paramètres de transmission 42
2.9 Critique 43
3 E-ADR 44
3.1 E-ADR pour les modèles de mobilité connus 44
3.1.1 Techniques de localisation 44
3.1.2 Prédiction de la prochaine position -Régression linéaire- . . 46
3.1.3 Modèle d'allocation de configuration 47
3.2 Extension d'E-ADR pour les modèles de mobilité inconnus « VHMM-based E-ADR » 47
3.2.1 Variable HMM-(2,1) 48
3.2.2 Validation du modèle VHMM-(2,1) 51
3.2.3 Allocation de configuration 53
4 Conclusion 53

III Évaluation expérimentale d'E-ADR 55
1 Conditions d'évaluation 55
2 Évaluation de E-ADR dans le cas de noeuds statiques et des obstacles mobiles 58
2.1 Scénario de test 58
2.2 Évaluation de performance 59
2.2.1 Évaluation du PLR 59
2.2.2 Évaluation du ToA et de la consommation énergétique . . . 62
3 Évaluation d'E-ADR dans le cas de noeuds mobiles à trajectoires connues . . 64
3.1 Scénario de test 64
3.2 Cas 1 : Utilisation des messages confirmés - retransmissions autorisées - 66
3.2.1 Évaluation du PLR 66
3.2.2 Évaluation du ToA et de la consommation énergétique . . . 69

3.2.3 Évaluation du délai aller-retour (RTD) moyen 72
3.2.4 L'impact des retransmissions 72
3.3 Cas 2 : Utilisation des messages non confirmés - Pas de retransmissions - 74
3.3.1 Évaluation du PLR 75
3.3.2 Évaluation du ToA et de la consommation énergétique . . . 76
3.3.3 Évaluation du délai de transmission 79
4 Évaluation de VHMM-based E-ADR : le cas de noeuds mobiles à trajectoires inconnues 80
4.1 Scénario de test 80
4.2 Évaluation de performance 81
4.2.1 Évaluation du PLR 81
4.2.2 Évaluation du ToA et de la consommation énergétique . . . 84
4.2.3 Évaluation du délai aller-retour (RTD) moyen 86
4.2.4 L'impact des $ADRACKLIMIT$ retransmissions 87
5 Conclusion 88

IV Optimisation des techniques d'accès dans LoRaWAN 89
1 Introduction 89
2 Approche de Duty Cycle dynamique 90
2.1 Mécanisme de partage du Duty Cycle 91
2.2 Principe du Duty Cycle dynamique 92
2.2.1 Gestion du partage du Duty cycle côté noeud émetteur . . . 93
2.2.2 Gestion du partage du Duty cycle côté serveur 93
2.2.2.1 Réponse du serveur à un noeud demandeur 95
2.2.2.2 Réponse du serveur au noeud donneur 96
2.3 Évaluation de la stratégie de partage de Duty Cycle (DC dynamique) 96
2.3.1 Scénario avec Duty Cycle fixe (S1) 96
2.3.2 Scénario avec Duty Cycle dynamique (S2) 97
2.3.3 Duty Cycle dynamique - gestion de l'allocation 98
2.3.3.1 Priorité au faible trafic (S3) 98
2.3.3.2 Priorité au plus faible niveau d'énergie et au noeud le moins distant (S4) 99
2.3.4 Comparaison DC dynamique / fixe 100
2.4 Duty Cycle dynamique combiné à E-ADR pour de meilleures performances 102
2.4.1 Description du fonctionnement de la combinaison 102
2.4.1.1 Définition des informations des noeuds (Type, T_{Dem}/T_{Don}) 102
2.4.1.2 Mise à jour des paramètres de partage 102
2.4.1.3 Traitement au niveau serveur pour envoyer un acquittement 103
2.4.2 Effet du DC Dynamique sur l'adaptation du débit dans un réseau de noeuds fixes 104
2.4.2.1 ADR Basique 104
2.4.2.2 ADR Basique - DC dynamique 104
2.4.2.3 Évaluation des pertes 105

2.4.3 Effet du DC Dynamique sur l'adaptation du débit dans un réseau de noeuds mobiles . . . 105
2.4.3.1 E-ADR . . . 106
2.4.3.2 E-ADR - DC dynamique . . . 106
2.4.3.3 Évaluation des pertes . . . 107
3 Approche d'allocation triplement conjointe SF-Slot-Canal . . . 108
3.1 Mécanismes d'accès proposés pour LoRaWAN : état de l'art . . . 109
3.2 Allocation dynamique triplement conjointe . . . 112
3.2.1 Principe d'allocation Slot/Canal . . . 113
3.2.2 Détection de données supplémentaires . . . 115
3.3 Évaluation de performances . . . 116
3.3.1 Scénario de test . . . 116
3.3.2 Évaluation du taux de collision . . . 118
3.3.3 Évaluation de l'équité dans l'occupation des canaux "Load Balancing" . . . 121
3.3.4 Évaluation du ToA et de la consommation énergétique . . . 121
3.3.5 Évaluation du délai de transmission . . . 122
4 Conclusion . . . 123

V Étude de la vulnérabilité de E-LoRaWAN 124
1 Introduction . . . 124
2 Les caractéristiques de sécurité LoRaWAN . . . 124
2.1 Méthode OTAA . . . 125
2.2 Cryptage CounTeR (CTR) . . . 125
2.3 Authentification par MIC . . . 126
3 Les attaques contre les réseaux LoRaWAN pour une jointure OTAA . . . 127
3.1 Attaque d'écoute "Eavesdropping" . . . 127
3.2 Les attaques par brouillage "Jamming" . . . 127
3.3 Attaque par usurpation d'identité "Spoofing" . . . 128
3.4 Attaque par retournement de bits "Bit-flipping" . . . 128
4 Les contre-mesures proposées pour le protocole LoRaWAN . . . 130
5 Étude d'exemples d'attaques du réseau E-LoRaWAN . . . 131
5.1 Scénario 1 : Bit-flipping et Requested Time-based ID-Spoofing . . . 131
5.2 Scénario 2 : Bit-flipping et RSSI based ID-Spoofing . . . 132
6 Solution contre la vulnérabilité de E-LoRaWAN . . . 133
6.1 Le message « Join Request » crypté . . . 134
6.2 Offset Codebook Mode OCB-AES pour LoRaWAN . . . 135
7 Évaluation du schéma de sécurité proposé pour LoRaWAN . . . 137
7.1 Évaluation du délai de transmission . . . 137
7.2 Évaluation de la consommation énergétique . . . 138
8 Conclusion . . . 139

Conclusion générale et perspectives 140
Conclusion générale . . . 140

Liste des publications **143**

Bibliographie **144**

Table des figures

I.1 Couverture géographique des technologies d'accès sans fil [7] 6
I.2 Architecture du réseau LoRaWAN 20
I.3 Classes des noeuds 21
I.4 Structure de PHYPayload 25
I.5 Structure de MACPayload 25
I.6 Structure de FHDR 25
I.7 Structure de MHDR 26
I.8 Structure de FCtrl UL 26
I.9 Structure de FCtrl DL 26
I.10 Structure de FOpts 26
I.11 Format du Join Request 27
I.12 Format du Join Accept 27
I.13 Procédures de jointure et protocole de sécurité LoRaWAN 28

II.1 ordre γ du HMM 51
II.2 Les $l = 8$ POIs considérés 51
II.3 Trajectoire en Zigzag Vs trajectoire prédite (HMM-1, HMM-2, HMM-3, VHMM-(2,1)) 52

III.1 Courant d'alimentation pour le nœud SX1272 LoRa dans différents États à $U = 3V$ 57
III.2 Position des gateways et des noeuds - Scénario statique 59
III.3 Évolution du PLR pour D_1 60
III.4 Évolution du PLR pour D_2 60
III.5 Évolution du PLR pour D_3 60
III.6 Évolution du PLR pour D_4 60
III.7 Évolution du PLR pour D_5 61
III.8 Évaluation du ToA pour D_1 62
III.9 Évaluation du ToA pour D_2 62
III.10 Évaluation du ToA pour D_3 63
III.11 Évaluation du ToA pour D_4 63
III.12 Évaluation du ToA pour D_5 63
III.13 Position des gateways et des noeuds 65
III.14 modèle de mobilité « carré » (D_5) 66
III.15 modèle de mobilité « carré réduit » (D_3,D_4) 66
III.16 modèle de mobilité « Zigzag » (D_1,D_2) 66
III.17 L'évolution du PLR pour D_1 68

III.18 L'évolution du PLR pour D_5 . . . 68
III.19 L'évaluation du ToA pour D_1 . . . 71
III.20 L'évaluation du ToA pour D_2 . . . 71
III.21 L'évaluation du ToA pour D_3 . . . 71
III.22 L'évaluation du ToA pour D_4 . . . 71
III.23 L'évaluation du ToA pour D_5 . . . 71
III.24 L'impact de la variation du $ADRACKLIMIT$ sur le nombre de retransmission moyen N_{avgre} pour D_1 . . . 73
III.25 Impact de $ADRACKLIMIT$ sur le DC pour D_1 . . . 74
III.26 L'évolution du PLR pour D_1 . . . 75
III.27 L'évolution du PLR pour D_5 . . . 76
III.28 L'évaluation du ToA pour D_1 . . . 77
III.29 L'évaluation du ToA pour D_2 . . . 77
III.30 L'évaluation du ToA pour D_3 . . . 77
III.31 L'évaluation du ToA pour D_4 . . . 77
III.32 L'évaluation du ToA pour D_5 . . . 78
III.33 L'évolution du PLR pour D_1 . . . 83
III.34 L'évolution du PLR pour D_2 . . . 83
III.35 L'évolution du PLR pour D_3 . . . 84
III.36 L'évolution du PLR pour D_4 . . . 84
III.37 L'évolution du PLR pour D_5 . . . 85
III.38 L'évaluation du ToA pour D_1 . . . 86
III.39 L'évaluation du ToA pour D_2 . . . 86
III.40 L'évaluation du ToA pour D_3 . . . 86
III.41 L'évaluation du ToA pour D_4 . . . 86
III.42 L'évaluation du ToA pour D_5 . . . 86

IV.1 Traitement DC-Dynamique au niveau du noeud . . . 94
IV.2 Réception d'un paquet avec partage de DC au niveau du serveur . . . 94
IV.3 Réponse du serveur à une demande de duty cycle supplémentaire . . . 95
IV.4 La réception d'ACK au niveau du noeud . . . 96
IV.5 Gestion du Duty Cycle dans la transmission d'un noeud . . . 97
IV.6 Duty Cycle dynamique . . . 98
IV.7 Résultat du scénario S4 sur le terminal du serveur . . . 99
IV.8 Le nombre de noeuds satisfaits S1-S2 . . . 100
IV.9 Le nombre de noeuds satisfaits S4-S4″ . . . 100
IV.10 Mise à jour de l'état de participation au DC d'un noeud . . . 103
IV.11 Allocation de Slot/Canal par le serveur . . . 115
IV.12 Détection des données supplémentaires par le noeud . . . 117
IV.13 SF-ALOHA . . . 119
IV.14 SF-Slot-Canal . . . 120
IV.15 Évaluation du taux d'occupation des canaux . . . 121

V.1 Mode CTR Vs. Mode LoRaWAN-CTR . . . 126
V.2 Attaque Bit-flipping . . . 129

V.3 Scénario 1 : impact des attaques Bit-flipping et Priorité(ID)-Spoofing 132
V.4 Scénario 2 : impact des attaques Bit-flipping et RSSI(ID)-Spoofing 133
V.5 Join Request sécurisé 134
V.6 Mode OCB 136

Liste des tableaux

I.1 Comparaison des technologies LPWANs : SigFox, LoRa et NB-IoT 15
I.2 Impact de la variation des paramètres sur la performance du LoRaWAN . . 18
I.3 Table des codes DR fournis par la spécification LoRaWAN [40] 19
I.4 Table des codes DR fournis par la spécification LoRaWAN [40] 25

II.1 les modes de configuration pour TP=14dB [49] 35
II.2 Précision de la prédiction 52

III.1 Courant d'alimentation pour le nœud SX1272 LoRa pour les différents États à $U = 3Volts$ 57
III.2 Le PLR des noeuds 60
III.3 La consommation énergétique des noeuds (J) 62
III.4 Le PLR des noeuds 67
III.5 La première allocation de mode pour chaque noeuds 70
III.6 Consommation énergétique des noeuds (J) 70
III.7 Les mesures du RTD moyen (s) 72
III.8 L'impact du $ADRACKLIMIT$ (m) sur la consommation énergétique moyenne par paquet reçu (J) 72
III.9 l'impact du $ADRACKLIMIT$ (m) sur le PLR des noeuds 73
III.10 Le PLR des noeuds 75
III.11 Consommation énergétique des noeuds (J) 78
III.12 Délai de transmission moyen mesuré pour les 5 noeuds (s) 79
III.13 Allocation de mode pour chaque noeuds 81
III.14 PLR moyen sur les 5 tests effectués pour chaque D_j 82
III.15 Consommation énergétique par paquet reçu (J) 85
III.16 RTD moyen (s) 87
III.17 L'impact du $ADRACKLIMIT$ (m) sur la consommation énergétique moyenne par paquet reçu (J) 87
III.18 l'impact du $ADRACKLIMIT$ (m) sur le PLR des noeuds 88

IV.1 La taille du paquet « Join Request » et son ToA 100
IV.2 PLR des demandeurs pour S3 et S4 (dû au dépassement de DC) 101
IV.3 PLR des demandeurs (dû au dépassement de DC) 105
IV.4 Allocation des modes, consommation du ToA et PLR 106
IV.5 PLR des demandeurs (dû au dépassement de DC) 107
IV.6 caractéristiques des noeuds 118
IV.7 $T_{collision}$ moyen du réseau LoRaWAN expérimenté 120

IV.8 ToA consommé par tous les noeuds en utilisant les différentes combinaisons (ms) 122
IV.9 Consommation énergétique moyenne par paquet reçu (J) 122
IV.10 Délai de transmission moyen par paquet (TD) (ms) 122

V.1 Délai total de la procédure de jointure 137
V.2 Délais des différentes étapes de la procédure de jointure 138
V.3 Consommation énergétique de la procédure de jointure (J) 138

Liste des Abréviations

ABP Activation By Personnalisation
ACK Acknowledgement
ADR Adaptative Data Rate
AES Advanced Encryption Standard
ALOHA Additive Links On-line Hawaii Area
AOA Angle Of Arrival
AppEUI Application Unique Identiffier
AppSKey Application Session Key
BLE Bluetooth Low Energy
BW Bandwidth
Chirp Compressed High Intensity Radar Pulse
CR Coding Rate
CSS Chirp Spread Spectrum
CTR Counter
DBPSK Differential Binary Phase-Shift Keying
DC Duty Cycle
DR Data Rate
DevAddr Device Adress
DevEUI Device Unique Identifier
DOA Direction Of Arrival
DoS Denial of Service
E Energy
E-ADR Enhanced Adaptative Data Rate
E-LoRaWAN Enhanced Long Range Wide Area Network
ETSI European Telecommunications Standards Institute
EU European Union
FDMA Frequency-Division Multiple Access
FSK Frequency-shift keying
GPS Global Positioning System

HMM Hidden Markov Model
ISM Industrial, Scientific and Medical
IoT Internet Of Thing
IQL Indicator Quality Link
LPWAN Low Power Wide Area Network
LoRa Long Range
LoRaWAN Long Range Wide Area Network
LTE-M Long Term Evolution Cat M1
MAC Medium Access Control
MHDR Mac HeaDeR
MIC Message Integrity Code
NB-IoT NarrowBand Internet Of Things
NFC Near Field Communication
NwkSKey Network Session Key
OFDMA Orthogonal Frequency-Division Multiple Access
OTAA Over The Air Activation
PHY Physical
PLR Packet Loss Rate
PN Pseudo-Noise
QAM Quadrature Amplitude Modulation
QoS Quality Of Service
QPSK Quadrature Phase-Shift Keying
RFID Radio-Frequency Identification
RFU Reserved for Future Use
RSSI Received Signal Strength
SF Spreading Factor
SNR Signal over Noise Ratio
TDMA Time-Division Multiple Access
TDOA Time Difference Of Arrival
ToA Time of Arrival
TP Transmission Power
VHMM Variable Hidden Markov Model
WLAN Wireless Local Area Network
WNAN Wireless Neighborhood Area Network
WPAN Wireless Personal Area Network
WWAN Wireless Wide Area Network
WWAN Wireless Wide Area Network

Résumé

Parmi les technologies de transmission sans fil existantes, les LPWANs (Low Power Wide Area Network) attirent de plus en plus d'attention, notamment grâce à leur longue portée radio et leur faible consommation d'énergie. Cependant, chercher à minimiser la consommation d'énergie peut parfois compromettre la résilience de la transmission des données face à des perturbations de l'environnement (interférences, obstacles) et de la mobilité des objets connectés. Par ailleurs, la longue portée radio aussi impose une limitation du temps d'occupation de la bande de fréquence par chaque noeud (e.g. «duty cycle» limité à 1%).

Dans cette thèse, nous nous intéressons à la technologie LoRa/LoRaWAN. Grâce à ses nombreux paramètres configurables, LoRaWAN peut s'adapter potentiellement à des environnements et applications IoT très hétérogènes. Son mécanisme de débit adaptable ADR (Adaptive Data Rate) lui permet de converger vers une configuration «optimale» pour économiser l'énergie. Plus récemment, des applications impliquant des nœuds mobiles s'intéressent aussi à l'utilisation de LoRa/LoRaWAN. Néanmoins, nous avons constaté que ADR actuel ne s'accommode pas bien à des réseaux dynamiques en présence des obstacles mobiles ou nœuds mobiles, ceci à cause de sa convergence lente et le recours à de nombreuses retransmissions qui induisent à leurs tours une augmentation de la consommation de l'énergie.

La contribution principale de cette thèse est l'amélioration de la QoS (Qualité de Service) de LoRaWAN pour le support de la mobilité. Pour y parvenir, une nouvelle version ADR a été proposée, E-ADR, avec un nouveau paradigme basé sur la prédiction des déplacements du mobile et la définition de la configuration adéquate selon les seuils en RSSI (Received Signal Strength Indication), évitant ainsi de nombreuses retransmissions de paquets perdus en cas de configuration inadéquate. La nouvelle configuration contribue à diminuer le temps de convergence vers une configuration optimale, et par conséquent la surconsommation d'énergie et le risque du dépassement de la limite de «duty cycle» à cause des retransmissions. Afin de choisir des seuils RSSI appropriés, notre algorithme E-ADR intègre la prédiction de la mobilité basée sur l'apprentissage à travers un modèle de Markov caché pour des trajectoires inconnues.

Face au risque du dépassement de la limite de duty cycle, nous avons apporté dans cette thèse une deuxième contribution sur la distribution dynamique de quota en duty cycle parmi les nœuds d'un réseau LoRa privé, permettant de mutualiser ainsi l'ensemble des duty cycles entre les nœuds sous-consommateurs et sur-consommateurs.

Bien que le choix diversifié des paramètres LoRa (e.g., Facteur d'étalement SF, canaux fréquentiels) permette des transmissions simultanées, un déploiement dense peut encore engendrer des collisions. Notre troisième contribution porte donc sur l'ordonnancement des slots selon le principe du TDMA dynamique. L'ensemble de ces contributions donnant naissance à un nouveau protocole "E-LoRaWAN" est implémenté et évalué sur une plate-forme composée des nœuds Waspmote SX1272, STM32, et LoPy4 et des gateways LoRa SX1272

connectés à un réseau via Ethernet. L'évaluation expérimentale montre l'efficacité de notre solution pour supporter de la mobilité tout en fournissant un bon compromis entre la fiabilité de transmission et l'efficacité énergétique.

Abstract

Among the existing wireless transmission technologies, LPWANs (Low Power Wide Area Network) are attracting more and more attention, particularly due to their long radio range and low power consumption. However, aiming to minimise power consumption can sometimes compromise the resilience of data transmission when facing environmental perturbations (interference, obstacles) and the mobility of connected objects. Furthermore, long radio range also faces a limitation on the time of occupation of the frequency band by each connected object (e.g. «duty cycle» limited to 1%).

In this thesis, we focus on the LoRa/LoRaWAN technology. Based on its several configurable parameters, LoRaWAN can potentially adapt to very heterogeneous IoT environments and applications. Its Adaptive Data Rate (ADR) mechanism allows it to converge to an «optimal» configuration to reduce energy consumption. More recently, applications involving mobile nodes are also interested in the use of LoRa/LoRaWAN. Nevertheless, we have found that current ADR does not adapt well to dynamic networks in the presence of mobile obstacles or mobile nodes, due to its slow convergence and recourse to frequent retransmissions.

The main contribution of this thesis is the improvement of the QoS of LoRaWAN to support mobility. To achieve this, a new ADR version has been proposed, E-ADR, with a new paradigm based on the prediction of mobile movements and the definition of the adequate configuration according to RSSI thresholds, thus avoiding frequent retransmissions of lost packets in case of inadequate configuration. The new configuration helps to reduce the convergence time to an optimal configuration, and consequently the over-consumption of energy and the risk of exceeding the «duty cycle» limitation due to a high number of retransmissions. In order to select appropriate RSSI thresholds, our E-ADR algorithm integrates learning-based mobility prediction through a Hidden Markov model for unknown trajectories.

Facing the risk of exceeding the duty cycle limit, we presented a second contribution on the dynamic distribution of duty cycle quota among nodes in a private LoRa network, allowing to share all duty cycles between under- and over-consumer nodes.

Although the variety of LoRa parameters (e.g., Spread Factor SF, frequency channels) allows simultaneous transmissions, dense deployment can still lead to collisions. Our third contribution, therefore, deals with the scheduling of time slots according to the principle of dynamic TDMA.

All these contributions were implemented as «E-LoRaWAN», and evaluated on a platform composed of the Waspmote SX1272, STM32, and LoPy4 nodes and SX1272 gateways connected to a local server via Ethernet. The experimental evaluation shows the effectiveness of our solution in supporting mobility while providing a good compromise between transmission reliability and energy efficiency.

Introduction générale

Dans ce chapitre, nous introduisons le contexte de notre thèse ainsi que les objectifs que nous nous sommes fixés. Nous allons exposer les questionnements à l'origine du sujet de recherche ainsi que la problématique de recherche traitée. Enfin, nous terminons ce chapitre par la structure du manuscrit.

Contexte général

L'Internet des objets (IoT) est une technologie émergente développée à partir de la convergence des technologies sans fil, des systèmes de détection et l'Internet. De nos jours, cette technologie stimule le développement de nouvelles applications et est devenue essentielle dans le développement tel que les bâtiments intelligents, les soins personnalisés, les systèmes de transport intelligents, les villes intelligentes, l'agriculture intelligente, etc. Ces très vastes domaines d'applications ont attiré à la fois le marché et la communauté de recherche et de l'innovation vers le « Smart world » où l'idée est de connecter la population, les objets, les animaux avec les services utilisés quotidiennement.

Le nombre exponentiel d'objets connectés et le développement des environnements intelligents nécessitent l'interconnexion d'un grand nombre d'équipements hétérogènes et la gestion de l'évolutivité du réseau tout en garantissant une qualité de service (QoS) satisfaisante. Cependant, pour assurer cette QoS et répondre aux différents critères de l'IoT, l'architecture classique de l'Internet est jugée rigide et ne se prête pas à ces nouvelles exigences en matière de QoS. Une QoS de bout en bout optimale et satisfaisante dans l'IoT est définie par une longue portée, une faible consommation d'énergie et un bon rapport coût-efficacité. En effet, les gateways (ou passerelles) implémentent plusieurs protocoles de communication qui impactent directement sur l'amélioration ou la dégradation de la QoS du réseau IoT. Les technologies radio à courte portée largement utilisées dans les maisons intelligentes, par exemple, ne sont pas adaptées aux scénarios qui nécessitent une transmission à longue distance. D'autre part, les solutions basées sur les communications cellulaires peuvent fournir une plus grande couverture, en consommant beaucoup trop d'énergie. Par conséquent, les exigences des applications IoT ont favorisé l'émergence d'une nouvelle technologie de communication sans fil : le réseau étendu à faible puissance (LPWAN). Cette nouvelle technologie de communication gagne de plus en plus en popularité dans les milieux industriels et de la recherche en raison de ses caractéristiques de communication à faible puissance, couverture étendue, faible coût et déploiement à grande échelle, répondant aux critères de l'IoT. Parmi les différentes technologies LPWAN, nous nous intéressons à la technologie LoRa qui est de plus en plus répandue vu son accessibilité basée sur un code source ouvert dédié au domaine de la recherche. Le réseau LoRa permet de communiquer sur de longues distances, il est très efficace sur le plan énergétique

(plus de 10 ans de durée de vie de la batterie [1]) et est peu coûteux. Ces aspects prometteurs ont donné lieu à des études expérimentales récentes sur la performance de la technologie LoRa dans des environnements extérieurs et intérieurs. Cependant, différentes questions restent ouvertes : Est-ce que cette technologie est capable de répondre en même temps à toutes les exigences de l'IoT ? Pouvons-nous considérer le déploiement de cette technologie dans différents environnements, y compris la mobilité par exemple ? Quels sont les paramètres influençant la QoS et comment peut-on optimiser leur configuration pour une meilleure QoS avec une très faible complexité ? La réponse à ces questions est l'objectif de notre thèse.

Objectif et contributions

Un réseau LoRa est basé sur deux composantes, à savoir LoRa et LoRaWAN. LoRa correspond à la couche physique et plus précisément à la technique de modulation utilisée et LoRaWAN définit le protocole MAC. Un réseau LoRa est caractérisé par sa longue portée, sa faible consommation énergétique, et son faible coût. Il repose sur une architecture basée sur une topologie en étoile étendue, permettant des communications bidirectionnelles à faible débit. Par ailleurs, il opère dans la bande de fréquences sans licence ISM (Industrial, Scientific, and Medical), sur laquelle la durée de transmission par cycle pour chaque émetteur est limitée. Afin de maximiser le débit et de minimiser la consommation énergétique, le réseau LoRa offre plusieurs combinaisons de configuration pour ses transmissions, gérées à travers un mécanisme d'adaptation de débit ADR (Adaptive Data Rate).

Ainsi, LoRa est devenu un candidat intéressant pour l'IoT grâce à ses caractéristiques techniques, ses nombreuses configurations possibles et à son débit de données adaptatif (ADR). Néanmoins, malgré sa flexibilité en termes de configurations, LoRa souffre souvent d'un manque de fiabilité dans la réception des paquets [2] et doit être conforme à certaines restrictions, telles qu'un temps d'activité limité du canal (moins de 1% de Duty Cycle). En outre, la configuration optimale de son protocole MAC « LoRaWAN » pour obtenir une meilleure QoS, tout en respectant la limitation du Duty Cycle et une faible consommation d'énergie, reste un défi à cause des objectifs antagonistes entre l'augmentation de la QoS et la minimisation de la consommation énergétique. De plus, la technique d'adaptation de débit ADR implémentée actuellement ne considère pas le cas de mobilité et malgré les différents travaux de recherches effectués sur LoRa, aucune attention n'a été accordée ni à l'optimisation de la performance, ni à l'adaptation des paramètres de configuration dans un contexte de mobilité.

L'objectif de nos travaux de recherche est d'optimiser la QoS globale du réseau LoRaWAN (meilleure utilisation des ressources, réduction des pertes, etc.) tout en optimisant l'efficacité énergétique des noeuds qui est une contrainte importante pour les objets IoT. Nous nous intéressons plus particulièrement à des environnements mobiles où les approches statiques ne sont pas adéquates et nécessitent de nouveaux mécanismes pour prendre en charge l'adaptation plus rapide de débit et l'utilisation des ressources.

Notre première contribution s'est intéressée à l'étude du mécanisme d'adaptation de débit (ADR). Une étude comparative a été menée sur les différentes variantes proposées dans la littérature. Mais toutes ces propositions n'ont pas tenu en compte du critère de mobilité. C'est ce qui nous a poussé à proposer un nouveau mécanisme dynamique basé principalement sur des

solutions d'apprentissage et de prédiction. Dans un premier temps nous avons proposé E-ADR basé sur un algorithme de régression linéaire. L'idée principale est de rendre le mécanisme ADR dynamique en l'adaptant à un environnement de mobilité et d'obstacles. Une évaluation de performances de E-ADR en le comparant à d'autres variantes [3–5] a été réalisée et a fait l'objet d'une publication dans la conférence *IWCMC* 2019 et d'une extension dans la revue *Telecommunication Systems* 2020.

Comme E-ADR a été initialement proposé dans des environnements contrôlés avec des trajectoires de mobilité connues, une extension de ce mécanisme a été proposée pour prendre en charge des modèles de mobilité quelconques. Pour ce faire un modèle de Markov caché à deux niveaux (1,2) est proposé pour faire la prédiction des différents déplacements de noeuds afin de pouvoir calculer la meilleure configuration. Les résultats du nouveau mécanisme "VHMM-EADR" sont en cours de soumission à une conférence.

Notre deuxième contribution s'attaque à la limitation du cycle de service offert pour chaque noeud et qui peut être dans plusieurs cas la source de perte des paquets puisqu'ils ne peuvent plus être envoyés une fois le duty cycle épuisé. Dans ce contexte, nous avons proposé un mécanisme de gestion dynamique du Duty Cycle basé sur le principe de partage du Duty Cycle entre les différents objets enregistrés dans le réseau LoRaWAN privé. Pour cela, nous proposons aux objets peu actifs (Température, par exemple) de céder le restant de leur temps d'activité aux objets très actifs (Caméra capturant des images, par exemple), tout en considérant la non-privation de ces noeuds. Cette solution permet d'accorder plus de chances aux noeuds de transmettre toutes leurs données. Cette étude a été publiée dans la conférence $ADHOC-NOW$ 2018.

Les mécanismes d'adaptation de débit (E-ADR, VHMM-EADR) et de gestion du duty cycle (DC dynamique) fonctionnent bien dans des réseaux peu denses où nous avons rarement la situation de collisions avec l'allocation différenciée des facteurs d'étalement (SF : Spreading Factor) et de canaux de fréquence aux différents noeuds. Néanmoins dans le cas de déploiement très dense, les SF et canaux différenciés ne peuvent plus être garantis, conduisant à des collisions. Ce phénomène est dû à la technique d'accès aléatoire ALOHA utilisée par LoRaWAN. Ainsi notre troisième contribution a consisté à la proposition d'une technique d'accès déterministe basée sur l'allocation du canal et du slot de temps à utiliser par chaque noeud. De ce fait, un nouveau protocole E-LoRaWAN est proposé, basé sur l'approche d'allocation triplement conjointe "SF-Slot-Canal". L'idée principale est la planification de slots de transmission sur des canaux libres et la réadaptation de ces planifications en fonction de l'évolution du réseau et les changements des besoins des noeuds en utilisant la méthode TDMA dynamique. Les résultats de simulation ont montré que cette approche nous permet de garantir l'élimination des collisions tant que les trois ressources "SF-Slot-Canal" sont disponibles.

La dernière partie traitée dans notre thèse, consiste à donner une ouverture de notre sujet vers l'étude des aspects de sécurité liés à LoRaWAN en général et à E-LoRaWAN en particulier. Cette étude traite les vulnérabilités de E-LoRaWAN en présentant deux exemples d'attaques et proposant une solution de contre mesure. Dans cette partie, nous avons proposé un nouveau schéma de sécurité basé sur le cryptage AES-OCB. Ce dernier permet de changer le positionnement des données dans les paquets LoRa et de rajouter des fonctions HASH permettant de sécuriser les transmissions contre l'attaque Bit-flipping. De plus, ce schéma propose le cryptage du Join Request afin d'écarter la possibilité d'usurpation d'identité des noeuds lors de la procédure de jointure. Cette étude a été publiée dans la conférence

COMNET 2018.

Structure du manuscrit

La suite de ce manuscrit de thèse est organisée en cinq chapitres.

Le **chapitre I** présente dans un premier temps un aperçu des solutions technologiques de l'IoT et principalement les réseaux à faible puissance et à large couverture (LPWAN). Les caractéristiques et le fonctionnement de base de différentes normes LPWAN ont été abordé. Ensuite, la norme LoRa et son protocole MAC (LoRaWAN), sur lesquels repose cette thèse, ont été abordé en détail.

Le **chapitre II** présente d'abord les défis concernant le mécanisme d'adaptation de débit ADR proposé par LoRaWAN et les différentes propositions dans la littérature. Ensuite, nous présentons deux extensions dynamiques « E-ADR » améliorant grandement l'adaptabilité du protocole LoRaWAN au contexte de mobilité.

Le **chapitre III** évalue les deux versions E-ADR et les compare aux différentes variantes ADR proposées par la littérature et l'ADR de base (ADR Basique) en termes de taux de perte de paquets, de délai et de consommation énergétique.

Le **chapitre IV** expose deux problématiques. La première partie du chapitre présente la première problématique liée à la limitation de duty cycle imposée par l'ETSI, fixée à 1%. Celle-ci peut engendrer une limitation de QoS lorsqu'il s'agit de transmission de trafic plus dense dans le cas des applications plus importante. D'où la proposition d'une alternative. Dans la deuxième partie du chapitre, nous exposons la possibilité de faire face à un taux de collision significatif dans le cas d'un réseau dense. Ceci même en utilisant le mécanisme d'adaptation de débit dynamique (E-ADR). C'est dans ce contexte que ce chapitre propose une approche d'allocation de ressources triplement conjointe «SF-Slot-Canal» basée sur la technique TDMA dynamique.

Le **chapitre V** présente une solution sur l'axe de recherche de sécurité. Le but est d'étudier la vulnérabilité du protocole E-LoRaWAN et de proposer un schéma pour renforcer la sécurité de la version améliorée du protocole que nous proposons. L'objectif est d'améliorer la QoS en permettant à tous les noeuds de bénéficier de la dynamicité de nos approches en toute sécurité.

Nous conclurons par ce chapitre que d'autres défis restent à relever dans le cadre de la garantie d'un autre pilier du niveau de QoS qui est la sécurité et qui sera dans les perspectives de nos travaux de recherche.

Enfin, dans la partie **conclusions et perspectives**, nous présentons une synthèse des contributions développées dans le cadre de nos travaux de recherche, en apportant des perspectives à ce travail.

Chapitre I

Les technologies IoT & LoRaWAN

Avec l'émergence d'une grande variété d'applications visant à rendre la vie des personnes plus confortable et l'environnement plus intelligent, différentes technologies de communication principalement radio sont utilisées pour assurer la connexion des milliers d'objets à l'Internet donnant naissance à l'Internet des Objets ou IoT. L'objectif de ce chapitre est de présenter un aperçu des nouvelles solutions technologiques de l'IoT à faible puissance et à large couverture. Nous nous focaliserons par la suite sur la Technologie LoRa et son protocole MAC LoRaWAN, faisant l'objet de nos travaux de thèse, en justifiant ce choix technologique.

1 – Introduction

Le nombre de dispositifs utilisés dans notre quotidien et qu'on souhaite connecter à l'Internet a connu une croissance prononcée. Ce nombre est passé d'un million en 1992 à plus de 50 milliards cette année [6]. Graduellement, cette croissance engendre la construction de ce qu'on décrit comme le paradigme de l'IoT. L'IoT est utilisé dans de multiples domaines d'application incluant la santé, le transport, les maisons et cités intelligentes, l'agriculture ainsi que plusieurs autres domaines. L'IoT n'est pas une technologie unique, mais il s'agit d'une combinaison de capteurs, des dispositifs, des réseaux et des logiciels qui fonctionnent ensemble pour accéder à des données exploitables. Contrairement à d'autres technologies qui tournent autour d'une architecture, d'un type de dispositif ou d'une méthode de connexion prédominante, l'IoT est au coeur d'un assemblage de technologies compétitives qui diffèrent dans leurs caractéristiques et qui coexistent dans le monde de l'IoT pour créer une intelligence supplémentaire.

Il existe toute une série de technologies permettant la connectivité IoT. Les plus courantes comprennent les réseaux personnels sans fils "Wireless Personal Area Networks - WPAN" (comme le Bluetooth), les réseaux locaux sans fils "Wireless Local Area Networks - WLAN" (comme le Wi-Fi), les réseaux étendus à faible puissance "Low Power Wide Area Network - LPWAN" (comme LoRa et SIGFOX) et les réseaux cellulaires (comme la 2G, 3G et 4G). Ces différentes technologies sont positionnées selon leur portée dans la Figure I.1 [7] afin d'identifier la zone d'application de chacune de ces solutions techniques.

Les réseaux de proximité basés sur l'identification par radio fréquence (RFID) ou la communication en champ proche (NFC) sont les réseaux de communication de type "near-me area network" pour les dispositifs à proximité immédiate.

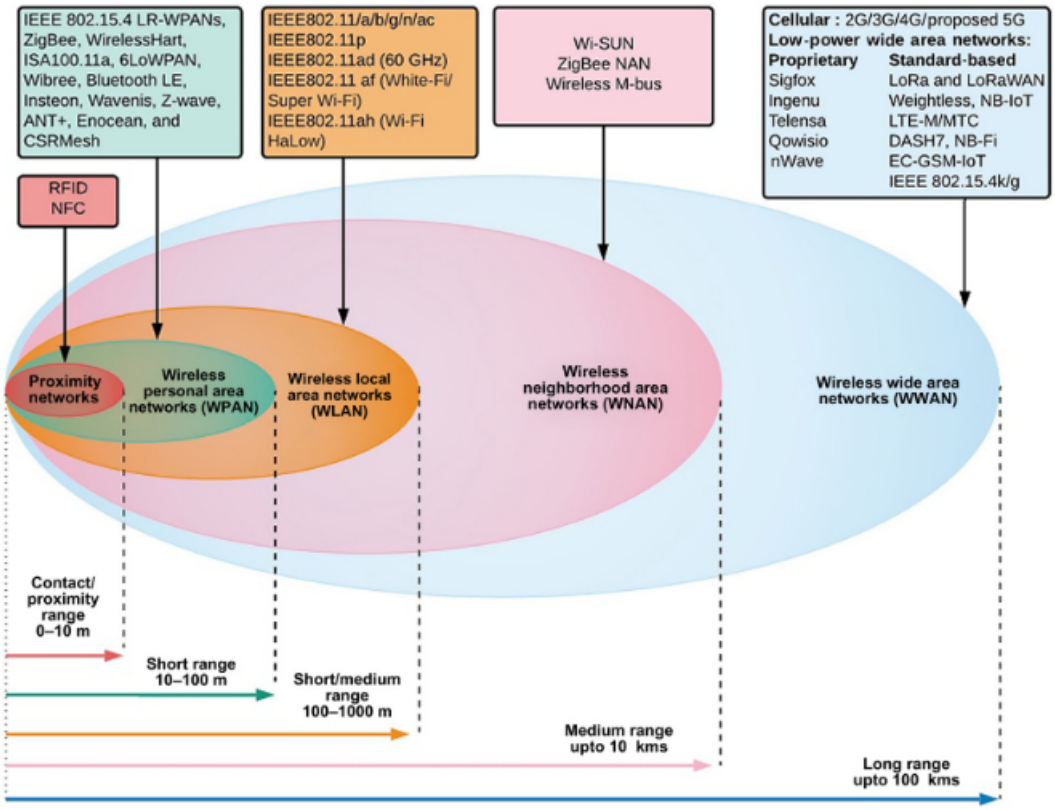


FIGURE I.1 – Couverture géographique des technologies d'accès sans fil [7]

Les WPAN sont utilisés pour transmettre des informations sur de courtes distances parmi le groupe de dispositifs participants ayant peu ou pas d'infrastructure. Ces réseaux peuvent être connectés à des plates-formes en nuage par l'intermédiaire d'un dispositif ou d'un serveur centralisé. La plupart des réseaux WPAN sont conçus pour un faible débit de données, une efficacité énergétique et une courte distance, et sont censés être des solutions peu coûteuses. Parmi les principales technologies WPAN, citons les réseaux à faible débit IEEE 80.15.4, ZigBee, WirelessHart, ISA100.11a, 6LoWPAN, Wibree, Bluetooth à faible consommation d'énergie (BLE), Insteon, Wavenis, Z-wave, ANT+, Enocean et CSRMesh.

Les réseaux WLANs sont principalement conçus pour l'échange de données à haut débit entre les appareils, avec une couverture du site limitée à quelques centaines de mètres. Les technologies WLAN comprennent les différentes variantes de la norme IEEE 802.11.

En outre, le réseau local sans fil "Wireless Neighbordhood Area Networks - WNAN" a évolué en tant que nouveau système architectural pour les applications de distribution locale sans fil à large bande, qui comprend une zone de service plus petite que les réseaux métropolitains, mais plus grande que les réseaux locaux. Il peut être utilisé pour des environnements résidentiels, de campus et de rue pour des applications de services publics et de réseaux intelligents (smart grids). Les technologies utilisées pour les WNAN sont Wi-SUN, ZigBee NAN et Wireless M-bus.

Les réseaux étendus sans fil ("Wireless Wide Area Networks - WWAN") sont conçus pour desservir des zones plus étendues que les WLAN et les WNAN. Ils répondent à de nouvelles exigences pour des applications différentes en termes de couverture, d'efficacité énergétique, de débit de données, d'évolutivité, de réutilisation des ressources, etc. Les WWANs peuvent être classés en deux grandes catégories : les réseaux cellulaires et les LPWANs. Les réseaux cellulaires tels que les 3G et 4G sont principalement conçus pour transférer des données à un débit élevé sur quelques kilomètres à plusieurs dizaines de kilomètres. Ces réseaux favorisent la mobilité et offrent donc une couverture étendue au-delà de la portée d'une seule cellule grâce à des mécanismes de transfert (roaming). Les LPWAN sont des technologies de communication sans fil conçues pour permettre des communications à longue distance avec une faible consommation d'énergie, une interface peu coûteuse et un débit binaire relativement faible adaptés aux applications IoT et M2M. On estime qu'un quart de l'ensemble des dispositifs IoT sera connecté à Internet au moyen de technologies LPWAN propriétaires ou standard [7]. Les solutions LPWAN impliquent généralement un grand nombre de dispositifs finaux, envoient des messages de taille limitée et tolèrent des délais de bout en bout relativement longs. Les technologies LPWAN complètent et parfois remplacent les technologies cellulaires et sans fil à courte portée traditionnelles en termes de performances pour diverses applications émergentes [7].

L'enjeu pour le développement des applications IoT est : une très basse consommation d'énergie permettant une grande autonomie de l'objet, un bas coût et une couverture très étendue. En comparant les différents protocoles de communication, le réseau LPWAN est le candidat regroupant les réseaux à basse consommation, longue portée et faible coût [8]. Contrairement aux autres types de technologies, ce dernier est destiné aux équipements à capacité de mémoire et puissance de calcul limitées pour lesquels une autonomie de plusieurs années est requise.

En effet, les technologies radio à courte et moyenne portée largement utilisées (WPANs, WLANs et WNANs) ne sont pas adaptées aux scénarios qui nécessitent une transmission à longue distance. Par ailleurs, les réseaux cellulaires traditionnels (1G, 2G, 3G et 4G) fournissent une plus grande couverture par rapport aux technologies WPANs, WLANs et WNANs, mais proposent des débits plutôt élevés et donc ils sont coûteux. Ainsi, pour certaines applications (ville intelligente, agriculture connectée, etc.), il s'agit de déployer des centaines de milliers de capteurs (monitoring de la qualité de l'air, gestion des déchets, ...) communiquant quotidiennement de faibles quantités de données, à faibles débits vers des serveurs sur Internet (cloud), et il n'est clairement pas envisageable d'utiliser les réseaux mobiles [9] pour de nombreux égards : coût, infrastructure, puissance requise, consommation énergétique, etc.

Depuis, une réponse adaptée aux équipements IoT ainsi qu'à leurs critères a fait son apparition sur le marché de la connectivité des objets et des capteurs communicants à moyenne et longue portée : les LPWANs. Ces derniers gagnent de plus en plus en popularité dans les milieux industriels et académique en raison de leur capacité à offrir une connectivité à faible consommation énergétique à un nombre considérable de terminaux, répartis sur de larges zones géographiques, avec un très faible coût.

2 – Les réseaux LPWANs

LPWAN permet une communication longue portée allant jusqu'à 10 à 40 km dans les zones rurales et de 1 à 5 km dans les zones urbaines [8]. En outre, il est très efficace sur le plan énergétique (plus de 10 ans de durée de vie de la batterie) [1] et peu coûteux, le coût d'un chipset radio étant inférieur à 2 euros et le coût de fonctionnement d'un appareil étant de 1 euro par an [7]. Ces aspects prometteurs du LPWAN ont donné lieu à des études expérimentales récentes sur la performance du LPWAN dans des environnements extérieurs et intérieurs [10–12]. En résumé, le LPWAN convient parfaitement aux applications IoT qui nécessitent la transmission de faible volume de données sur de longues distances. De nombreuses technologies LPWAN sont apparues dans la bande de fréquences sous licence ou non, notamment DASH7, Weightless, Sigfox, NB-IoT et LoRa, qui sont les plus répandues. Une partie des technologies LPWANs utilise des bandes de fréquences sous licence, c'est-à-dire que l'autorité d'un état réserve des bandes de fréquences pour ces technologies. Une seconde partie utilise les bandes de fréquences sans licence (le spectre non licencié) permettant l'accès à certaines bandes de fréquences à faible coût au dépit d'une plus grande probabilité d'interférence. Nous présentons dans ce qui suit quelques solutions LPWANs utilisées pour les applications IoT.

2.1 DASH7

Le protocole DASH7 Alliance (D7AP) est une norme de communication pour les réseaux de capteurs sans fil développée par DASH7 Alliance [13]. L'Alliance DASH7 est une société à but non-commercial mise en place pour le développement de la spécification DASH7. Ce protocole fonctionne dans une bande de fréquences inférieure à 1 GHz sans licence (433 MHz, 868 MHz, 915 MHz) et peut atteindre une portée de communication allant jusqu'à 2 km. La conception de DASH7 est basée sur la norme ISO1800-7 [1], utilisée pour l'identification des fréquences radio et inventée par le département de la défense des États-Unis. La conception de DASH7 est basée sur le concept de BLAST (Bursty-Light-Asynchrone-Stealth-Transitive). Le terme "bursty" désigne la transmission de courtes séquences de données sporadiques limitée à 256 octets, qui est considérée comme légère (light). La communication est asynchrone, et est basée sur une approche de commande/réponse sans synchronisation périodique. Elle est furtive (stealth) car les appareils n'ont pas besoin de balises de découverte périodiques pour pouvoir répondre en communication. Elle est définie comme transitive, parce que le DASH7 prend en charge la mobilité. Les appareils peuvent se déplacer dans différentes zones de couverture GW (Gateways). DASH7 fournit une solution de pile protocolaire complète pour le LPWAN où les nœuds peuvent établir la communication sans se soucier de la complexité du réseau MAC ou de la couche physique. La topologie de réseau par défaut utilisée par DASH7 est une topologie arborescente. Toutefois, elle offre également la possibilité d'utiliser une topologie en étoile, si nécessaire. Le protocole MAC pour DASH7 exige que les nœuds écoutent périodiquement sur le canal de communication, ce qui induit une augmentation de la consommation énergétique. Le protocole DASH7 dispose également d'un mode de réveil consommant un peu plus d'énergie [13]. Dans ce mode, le nœud envoie un beacon annonçant l'heure à laquelle il enverra les données. Le nœud remarque un signal au-dessus du niveau de bruit et enregistre l'horodatage auquel les données doivent être reçues. Le nœud se met

ensuite en veille jusqu'à ce que l'horodatage soit atteint lorsqu'il se réveille pour recevoir les données. Voici un résumé des avantages et des inconvénients de DASH7.

— Avantages : bonne pénétration contre les interférences pour l'environnement extérieur et intérieur ; offre une solution de pile protocolaire complète, flexibilité d'utilisation de la topologie en arbre ou en étoile ; faible latence du réseau.

— Inconvénients : la faible latence se fait au détriment de la durée de vie des batteries et de la complexité ; peu de solutions commerciales sont disponibles, car elles sont plus récentes que les autres technologies LPWAN, une consommation énergétique plus élevée par rapport aux autres LPWANs.

2.2 WEIGHTLESS-SIG

Weightless est à la fois le nom de la technologie et d'un groupe appelé Weightless Special Interest Group [14]. Il s'agit d'une norme propriétaire développée pour la communication de machine à machine. Le groupe a présenté trois normes LPWAN qui peuvent fonctionner sur le spectre sous licence ou sans licence. Chacune de ces normes a des caractéristiques, une consommation d'énergie et une portée de communication différentes. Alors que toutes les normes utilisent la cryptographie à clé symétrique pour l'intégrité et l'authentification du réseau. Voici un résumé des différentes normes ainsi que les avantages et les inconvénients de cette technologie.

Weightless-W : cette norme présente une meilleure propagation du signal par rapport aux deux autres normes. La norme Weightless-W peut transmettre des paquets de données d'une taille maximale de 10 octets à des débits compris entre 1 Kbps et 10 Mbps. La norme Weightless-W prend en charge un large éventail de facteurs de propagation et de schémas de modulation multiples comme la DBPSK (Differential BPSK) et la 16-QAM (16-Quadrature Amplitude Modulation). Afin de prolonger la durée de vie de la batterie, les nœuds terminaux communiquent avec la station de base via une bande étroite et des niveaux de puissance plus faibles que ceux des stations de base. L'inconvénient de Weightless-W est que l'accès partagé aux espaces TV blancs n'est accordé que pour des régions limitées. Pour remédier à cela, Weightless SIG introduit Weightless-P et Weightless-N qui sont disponibles pour un accès partagé à l'échelle mondiale.

Weightless-N : Cette norme est la plus efficace en termes de consommation d'énergie par rapport à Weightless-W et à Weightless-P. Il s'agit d'une norme à bande ultra étroite qui n'offre qu'une communication unidirectionnelle entre les appareils terminaux et la station de base. Cette norme utilise le schéma de modulation par déplacement de phase binaire différentiel. L'inconvénient de la norme Weightless-N est qu'en raison de sa caractéristique de communication unidirectionnelle, elle peut être utilisée avec un nombre restreint d'applications qui ne nécessitent pas de communication bidirectionnelle.

Weightless-P : Elle offre une communication bidirectionnelle avec une transmission de données entièrement reconnue pour sa fiabilité et des possibilités de mise à jour du micrologiciel par voie aérienne. Elle prend en charge des données allant jusqu'à 48 octets et peut atteindre des débits de données de 0,625 kbps à 100 kbps. Weightless-P peut fonctionner dans n'importe quelle bande de fréquences, mais elle est actuellement définie pour fonctionner

uniquement dans les bandes de fréquences inférieures à 1GHz exemptées de licence. Elle utilise la modulation par déplacement de phase en quadrature (QPSK) et la modulation par déplacement minimal gaussien (GMSK) pour moduler les signaux.

— Avantages : la portée de la communication est de 2 à 5 km et plus ; grâce à la démodulation avancée, il peut fonctionner avec les technologies de radio-fréquence existantes avec un minimum d'interférences ; les modèles Weightless-W et Weightless-P offrent une communication bidirectionnelle ; les capteurs de faible débit peuvent utiliser le modèle Weightless-N tandis que le modèle Weightless-P offre un débit adaptatif.

— Inconvénients : Weightless nécessite un spectre TV en espace blanc pour fonctionner en Weightless-W. Selon la norme utilisée, l'autonomie de la batterie peut aller jusqu'à plus de 2 ans, mais reste tout de même faible par rapport aux autres LPWANs.

2.3 Sigfox

Sigfox [15] est un type de technologie cellulaire qui fournit des solutions sur-mesure permettant aux appareils sans fil de se connecter à une station de base propriétaire en utilisant une connexion IP à très faible puissance et à faible débit de données. Il s'agit d'une technologie propriétaire développée et maintenue par une société française, Sigfox, qui utilise la technique de modulation BPSK pour la transmission. Il s'agit d'un signal à bande ultra étroite (petits morceaux de 100 Hz) et les données sont codées en changeant la phase de l'onde porteuse, ce qui permet au récepteur de recevoir dans de petites tranches de spectre, qui réduit l'effet du bruit, augmentant ainsi sa portée et réduisant la consommation d'énergie. Sigfox utilise les bandes de fréquences ISM pour la communication. Il fonctionne à 868 MHz en Europe et à 902 MHz aux États-Unis. Selon Sigfox, un million d'appareils terminaux peuvent être connectés à un seul point d'accès et peuvent assurer une couverture de 3 à 10 km dans les zones urbaines à un débit de 100 bps et de 30 à 50 km dans les zones rurales. Les faibles débits binaires de transmission augmentent la latence de la communication et la rendent sensible aux interférences avec d'autres technologies utilisant la même bande. SigFox prend des fragments de spectre très étroits (100 Hz, ce qui donne 8000 canaux) [16] et change la phase de l'onde radio porteuse pour coder les données. Cela permet au récepteur de n'écouter que dans un minuscule fragment de spectre, atténuant l'effet du bruit. Sigfox envoie chaque message trois fois sur des fréquences de canal différentes, en s'assurant qu'il est reçu par au moins une des stations de base, ce qui confère une grande fiabilité à la liaison montante. Sigfox peut envoyer 140 messages en liaison montante de 12 octets maximum et peut recevoir 4 messages en liaison descendante de 8 octets par jour. Compte tenu du faible débit de données et de la latence élevée, Sigfox est adapté aux applications qui ont besoin de faibles débits de transmission. Étant donné qu'il s'agit d'une technologie propriétaire et fermée, les chercheurs externes ne disposent que de peu de liberté pour innover dans ce domaine. Voici un résumé des avantages et des inconvénients de Sigfox.

— Avantages : faible puissance nécessaire en raison de l'absence de circuits de réception ; la modulation lente permet d'obtenir une portée plus élevée, ce qui la rend plus adaptée aux applications simples.

— Inconvénients : ce n'est pas une norme à code source ouvert ; il n'offre que des communications en liaison montante ; les interférences radio sont importantes ; il offre une faible sécurité [16].

2.4 NB-IoT

NarrowBand IoT (NB-IoT) est une technologie radio à bande étroite LPWAN développée et normalisée par le 3GPP (3rd Generation Partnership Project) [17]. Cette norme utilise les bandes de communication cellulaire pour connecter les appareils IoT. Elle est l'une des nombreuses technologies de l'Internet mobile des objets (MIoT) conçues et normalisées par le 3GPP. Il existe trois modes de fonctionnement pour NB-IoT : autonome, in-band, guard-band [17].

En mode autonome, le signal NB-IoT occupe toute une plage de 200 kHz du signal de la porteuse GSM. Dans les modes en bande (In-band) et en bande de garde, le NB-IoT est implémenté comme un bloc de ressources physiques (PRB) de 180 kHz à l'intérieur du signal porteur LTE. Le NB-IoT réduit les fonctionnalités du protocole LTE au minimum et les modifie pour s'adapter aux cas d'utilisation de l'IoT. Une fois cette modification effectuée par le NB-IoT aux fonctionnalités LTE, elle se fait avec le système back-end qui est utilisé pour envoyer des informations aux noeuds. Comme la diffusion consomme de l'énergie sur batterie, ce qui est essentiel dans le cas des noeuds IoT, la fréquence d'envoi des données et leur taille sont réduites au minimum. La communication est optimisée en fonction de l'objectif de l'IoT et des caractéristiques telles que l'agrégation de porteuses ou la double connectivité qui n'est pas nécessaire pour les noeuds IoT sont évitées. NB-IoT utilise la modulation QPSK [18] et utilise l'accès multiple par répartition en fréquence orthogonale (OFDMA) pour la transmission en liaison descendante et l'accès multiple par répartition en fréquence (FDMA) pour la communication en liaison montante. La taille maximale des paquets de données pour NB-IoT est de 1600 octets avec un débit de données sur la liaison montante de 20 kbps et un débit de données sur la liaison descendante de 200 kbps. Comme indiqué dans [19], en transmettant à un débit de 200 octets par jour, NB-IoT peut avoir une durée de vie de la batterie allant jusqu'à 10 ans. Voici un résumé des avantages et des inconvénients du NB-IoT.

— Avantages : durée de vie de la batterie de plus de 10 ans ; plus de 100 000 appareils par cellule ; prise en charge des fonctions LTE telles que la localisation, la sécurité et l'authentification.

— Inconvénients : l'accusé de réception limité des messages en raison de la capacité de la liaison descendante ; l'augmentation de la latence due à l'agrégation des paquets ; les faibles performances lorsque le réseau est soumis à un trafic de données important ; la technologie est très récente par rapport à d'autres technologies plus matures [7].

2.5 LoRa/LoRaWAN

LoRa est une technologie de couche physique LPWAN développée et brevetée par Semtech Corporation [20]. LoRa utilise sa technique exclusive de modulation à spectre étalé [8], inventée initialement par une start-up française Cycleo puis acquise par Semtech en 2012. Elle fonctionne dans la bande de fréquence ISM (Industriel, Scientifique et Médical : UE 433 MHz 868 MHz, Asie 430 MHz et USA 915 MHz). Dans le but d'avoir des transmissions simultanées dans le même canal, LoRa se base sur la technique de modulation "chirp spread spectrum" (CSS) qui répartit un signal à bande étroite sur une bande passante plus large, permettant de réaliser une communication bidirectionnelle.

Le protocole de communication LoRaWAN appartient à la deuxième couche du modèle OSI qui est la couche MAC (Media Access Control Layer). En plus de la grande portée avec une faible consommation d'énergie, cette technologie offre plusieurs fonctionnalités telles que la communication bidirectionnelle, une sécurité de bout en bout grâce au cryptage AES, un enregistrement "On-The-Air" (OTAA) des noeuds, etc.

Les réseaux LoRaWAN sont mis en œuvre selon une topologie en étoile qui se compose de noeuds et de passerelles (gateways) qui relaient les messages des noeuds d'extrémités à un serveur réseau à travers la technologie IP. Ce dernier communique à son tour avec un serveur d'application. La spécification LoRaWAN [21] publiée par LoRa Alliance définit trois types de service : Classe A, classe B et classe C. Tout noeud LoRaWAN utilisé dans le réseau doit mettre en œuvre l'une de ces trois fonctionnalités. Dans la plupart des cas, les noeuds implémentent la classe A grâce à sa faible consommation énergétique, tandis que les classes B et C sont facultatives puisqu'elles consomment plus d'énergie [22].

2.6 Synthèse et choix technologique

Toutes ces technologies sont prometteuses pour le domaine de l'IoT. Par conséquent, lors du choix d'une technologie LPWAN appropriée pour une application IoT, de nombreux facteurs doivent être pris en compte, notamment la QoS, l'autonomie, la latence, la scalabilité, la taille de la charge utile (data payload), la portée radio, le déploiement déjà existant (témoignant la maturité de la technologie et disponibilité des composants) et le coût. Dans la suite, nous comparons les technologies LPWAN les plus répandues, LoRaWAN, SigFox et NB-IoT selon ces différents critères et nous justifions notre choix de LoRaWAN pour la suite de nos travaux de recherche.

— **Portée et couverture :**

Le principal avantage de LoRa est qu'une ville entière peut être couverte par une seule station de base. LoRa se concentre principalement sur la catégorie de dispositifs qui sont installés dans des endroits éloignés de la portée typique des réseaux cellulaires. Le déploiement de NB-IoT est limité aux stations de base 4G/LTE. Il n'est donc pas adapté aux régions rurales ou suburbaines qui ne disposent pas d'une couverture 4G. Un avantage important de LoRaWAN est sa flexibilité. Par ailleurs, LoRaWAN peut avoir une couverture de réseau plus large que le réseau NB-IoT. La portée de NB-IoT et du réseau LoRaWAN et Sigfox sont indiquées dans la Table I.1.

— **Durée de vie :**

Dans les réseaux SigFox, LoRa et NB-IoT, les noeuds sont en mode veille la plupart du temps en dehors du fonctionnement, ce qui réduit la quantité d'énergie consommée, augmentant ainsi la durée de vie des noeuds. Dans LoRaWAN, les noeuds peuvent dormir aussi peu ou aussi longtemps que l'application le souhaite, car il s'agit d'un protocole asynchrone,

basé sur ALOHA. Cependant, dans NB-IoT, en raison d'une synchronisation régulière, le noeud consomme une énergie supplémentaire. Ces demandes d'énergie supplémentaires déterminent que la durée de vie de la batterie du noeud NB-IoT est plus courte que celle des noeuds basés sur LoRa ou Sigfox.

— **Débit et latence :**

Le réseau NB-IoT bénéficie d'une faible latence et d'un débit de données élevé (200 kbps) comparé à LoRa (50 kbps) ou Sigfox (100 bps). D'autre part, contrairement à SigFox, LoRa offre une classe C pour gérer également une faible latence bi-directionnelle au détriment de plus de consommation énergétique. Pour les applications qui sont insensibles à la latence et qui n'ont pas une grande quantité de données à envoyer, SigFox et LoRa de classe A sont les meilleures options. Pour les applications qui nécessitent une faible latence, NB-IoT et LoRa de classe C sont les meilleurs choix. En résumé, LoRa peut être utilisée dans les deux cas grâce aux différentes classes disponibles.

— **Scalabilité** :

La prise en charge du nombre massif de noeuds est l'une des principales caractéristiques de SigFox, LoRa et NB-IoT. Ces technologies fonctionnent bien avec le nombre croissant des noeuds connectés. Plusieurs techniques sont envisagées pour supporter le facteur d'échelle, comme l'exploitation efficace de la diversité dans un canal, ainsi que dans le temps et l'espace. Toutefois, NB-IoT offre l'avantage d'une plus grande extensibilité que SigFox et LoRa. NB-IoT permet la connectivité de dispositifs jusqu'à 100 K par cellule, contre 50 K par cellule pour SigFox et LoRa [23].

— **Taille de la charge utile par paquet** :

NB-IoT offre l'avantage d'une longueur maximale de charge utile par paquet. NB-IoT permet la transmission de données jusqu'à 1600 octets. LoRa permet d'envoyer un maximum de 250 octets de données. Par ailleurs, SigFox propose la plus petite longueur de charge utile de 12 octets, ce qui limite son utilisation sur diverses applications de l'IoT qui ont besoin d'envoyer des données de grande taille. En résumé, le choix technologique dépend aussi de la taille maximale de données (charge utile) selon le besoin de l'application IoT.

— **Déploiement (maturité) :**

Les spécifications de NB-IoT ont été publiées en juin 2016, il a fallu donc plus de temps pour établir son réseau, alors que SigFox et LoRa sont arrivés à un degré de déploiement élevé et sont actuellement commercialisés dans divers pays et villes. LoRa présente l'avantage de pouvoir être actuellement déployée dans 42 pays, contre 31 pour SigFox [24, 25]. Néanmoins, les déploiements mondiaux de LoRa et SigFox sont toujours en cours, et l'un des avantages

majeurs de LoRa est sa flexibilité. Contrairement à SigFox et NB-IoT, LoRa peut être déployé en privé ou par des opérateurs. C'est-à-dire un réseau local utilisant une gateway LoRa ainsi qu'un fonctionnement de réseau public via des stations de base. Dans le domaine industriel, un modèle d'exploitation hybride pourrait être utilisé pour déployer un réseau LoRa local dans les zones d'usine et utiliser le réseau LoRa public pour couvrir les zones extérieures.

— **Qualité de service :**

SigFox et LoRa utilisent des spectres sans licence et des protocoles de communication asynchrones. Ils peuvent subir les interférences, trajets multiples et évanouissements. Toutefois, ils ne peuvent pas offrir la même qualité de service que celle fournie par NB-IoT. NB-IoT utilise un spectre sous licence et un protocole synchrone basé sur le LTE, qui sont optimaux pour la qualité de service au détriment du coût (e.g., les prix de spectre LTE sous licence s'élèvent à plus de 500 millions d'euros par MHz [26]). En raison de la qualité de service et du compromis des coûts, NB-IoT est préféré pour les applications qui exigent une qualité de service garantie, tandis que les applications qui n'ont pas cette contrainte peuvent choisir LoRa ou SigFox.

En résumé, le choix de l'une des technologies SigFox, LoRa et NB-IoT dépend des besoins des applications et du coût d'investissement. Table I.1 résume les caractéristiques de chacune de ces technologies.

Pour à la fois, ses avantages de déploiement public/privé, son code source ouvert, sa grande couverture à un débit adaptable, une consommation énergétique très faible et sa possibilité d'envoi de paquets de taille moyenne par rapport à NB-IoT et remarquable par rapport à SigFox, nos travaux de thèse se sont orientés vers la technologie LoRa et l'amélioration de sa qualité de service. Le but est de proposer des solutions permettant d'avoir de meilleures performances et d'augmenter la capacité de LoRaWAN à s'adapter aux applications exigeant de la QoS différenciée. Dans la suite, nous détaillons la technologie LoRa/LoRaWAN, ses caractéristiques, sa spécification, ses restrictions et ses mécanismes de gestion de ressources.

Table I.1. Comparaison des technologies LPWANs : SigFox, LoRa et NB-IoT

	SigFox	LoRa/LoRaWAN	NB-IoT
Standard	Compagnie SigFox	LoRa-Alliance	3GPP
Autonomie	5 à 10 ans	10 ans	< 5 ans
Schéma de modulation	DBPSK ou GFSK	CSS	DL :OFDMA UL :SC-FDMA
Fréquence	Bandes ISM sans licence	Bandes ISM sans licence	Bande LTE avec licence
Bande passante	100 Hz	125kHz, 250kHz et 500kHz	200 kHz
Débit maximal	100 bps	50 Kbps	DL :250 kbps UL :200 kbps
Bi-directionnel	Limité	Oui	Oui
Nombre max de messages par jour	UL : 140 et DL :4	illimité	illimité
Taille de la charge utile (Bytes)	UL :12 et DL :8	dépend du débit (maximum 250 octets)	1600
Latence	10 sec	10 sec	<10 sec
Topologie	étoile	étoile	étoile
Capacité	50000 noeuds	50000 noeuds	$>$ 50000 noeuds
Portée	10 Km (urbain), 45 Km (rural)	10 Km (urbain), 30 Km (rural)	8Km (urbain), 25 Km (rural)
Immunité contre les interférences	Grande	Grande	faible
Mécanisme de sécurité	Brouillage, utilisation des clés privées	Cryptage AES128	cryptage LTE
Adaptation de débit	Non	Oui	Non
Réseau privé autorisé	Non	Oui	Non
Références	[28–30], [32]	[29–32]	[27, 28], [30], [32]

3 – LoRa/LoRaWAN

3.1 Modulation Chirp Spread Spectrum (CSS) dans LoRa

Comme son nom l'indique, LoRa (Long-Range) fournit une connectivité fiable et de longue portée en utilisant une technique d'étalement du spectre « Chirped Spread Spectrum (CSS) » avec un large spectre. La couche physique LoRa est sous licence et brevetée par la société Semtech. Elle utilise la modulation CSS grâce à laquelle le signal est réparti sur une bande de fréquence plus large (ISM) en créant une altération linéaire de la fréquence [33] améliorant considérablement la sensibilité du récepteur. La modulation CSS consiste à moduler

les bits par des chirps. Dans la modulation CSS, l'étalement est obtenu en générant un signal chirp qui varie continuellement en fréquence. L'avantage de cette méthode est que les décalages de synchronisation et de fréquence entre l'émetteur et le récepteur sont équivalents, ce qui réduit considérablement la complexité de la conception du récepteur. Cet étalement permet de résister aux interférences et rend LoRa robuste au bruit de canal. Le maintien de ces chirps dans un spectre d'étalement plus large réduit également l'évanouissement par trajets multiples (multi-path fading) et améliore donc les performances dans les environnements urbains [23]. Les techniques de modulation CSS se distinguent par leur faible consommation d'énergie et leur résistance aux dégradations des canaux.

Le signal émis par LoRa (modulé en « chirp ») est formé de plusieurs symboles. Chaque symbole est représenté par (N= 2^{SF}) « Chips » (Un Chip représente en fait des impulsions d'un code à spectre étalé, tel qu'une séquence de code de pseudo-bruit (PN)) et contient SF bits, c'est-à-dire que si la transmission utilise SF8 (SF=8), un symbole contiendra 8 bits et est représenté par 2^8 chips. Ainsi, plus la valeur de SF est élevée, plus le nombre de chips utilisés pour représenter un symbole est important. La durée de transmission d'un symbole est de T_s secondes inversement proportionnelle à la bande passante BW (Eq (I.1)). Le débit de symbole D_s est donc représenté par Eq (I.2). En toute logique, plus la bande passante est élevée, plus le débit des symboles sera élevé. Comme chaque symbole comprend SF bits, alors le débit binaire D_b est défini par Eq (I.3).

$$T_s = \frac{2^{SF}}{BW} \qquad \text{(I.1)} \qquad\qquad D_s = \frac{1}{T_s} \qquad \text{(I.2)} \qquad\qquad D_b = SF * D_s \qquad \text{(I.3)}$$

Cette technique de modulation offre une variété de vitesses de données pour diverses BW. Dans les réseaux LoRa, le débit de données est choisi en fonction des besoins en termes de portée. Cette capacité de moduler différents taux d'échantillonnage donne à LoRa l'avantage de pouvoir décoder de nombreux signaux en même temps. Ces différents taux d'échantillonnage sont connus sous le nom de « facteurs d'étalement - Spreading factors (SF) ». Ces SFs permettent de décrire l'ampleur de l'augmentation d'occupation de la bande spectrale ainsi que le gain de codage des systèmes à étalement de spectre.

3.2 Caractérisation des paramètres de transmission LoRa

Pour permettre des transmissions simultanées, les nœuds LoRa sont configurés en utilisant des paramètres distincts : facteur d'étalement (SF), largeur de bande (BW), taux de codage (CR) et la puissance de transmission (TP) [21], ce qui permet plusieurs possibilités de configuration (6720 combinaisons possibles [34]). Les performances du réseau LoRaWAN dépendent largement de la configuration de ces paramètres. Chaque ensemble de paramètres a un effet direct sur le débit (DR), le temps de transmission (ToA), la sensibilité en réception définie par le rapport signal sur bruit (SNR) ou par l'indicateur de force du signal reçu (RSSI) (plus le RSSI « Received Signal Strength Indication » est élevé, plus la sensibilité en réception est bonne) et par conséquent la consommation d'énergie (E).

3.2.1 Facteur d'étalement (SF)

LoRa compte 6 SF différents : 7, 8, 9, 10, 11, 12. Les communications radio avec différents SF sont orthogonaux les unes aux autres, ce qui permet de séparer les transmissions en utilisant différents SFs [35]. Ce paramètre est pertinent pour la technique d'étalement du spectre. Dans le spectre étalé, plus la valeur de ce paramètre est élevée plus le récepteur est capable de réduire le bruit du signal. Ainsi, plus la valeur choisie est élevée, plus la durée de transmission est longue, mais aussi plus la portée radio est grande puisque la sensibilité du récepteur est meilleure. En outre, les transmissions simultanées excluant l'orthogonalité entraînent une perte de tous les paquets (collision), sauf si l'une de ces transmissions utilise une puissance de transmission (TP) beaucoup plus élevée conformément à la condition SNR (rapport signal sur bruit) (le signal reçu a au moins 6 dB de plus que les autres signaux) [36]. Celui-ci éliminera toutes les autres transmissions et sera le seul à être reçu. Ce phénomène est appelé « effet de capture » et a été étudié dans [37].

3.2.2 Bande passante (BW)

La valeur de la bande passante (BW) indique la largeur de bande utilisée pour le signal de transmission. Dans [38], il existe 8 canaux définis dans la bande de 868 MHz de l'Europe avec trois différentes bandes passantes possibles : 125 KHz, 250 KHz et 500 KHz. Si une transmission rapide est nécessaire, une valeur de 500 KHz est préférable. Mais si une portée plus longue est nécessaire, la valeur de 125 KHz doit être configurée. Moins la largeur de bande, plus le temps de transmission est long, mais aussi meilleure la sensibilité (mesurée par RSSI).

3.2.3 Taux de codage (CR)

LoRa utilise le code correcteur d'erreur FEC pour améliorer la robustesse de la connexion sans fil. Ce type de codage se traduit par des bits supplémentaires dans la charge utile de la couche physique de LoRa, qui est contrôlée par le paramètre CR. Le modem LoRa utilise le CR pour assurer une protection accrue contre les salves d'interférences et les erreurs de décodage. LoRa permet aux paramètres CR d'être réglé à $\frac{4}{5}$, $\frac{4}{6}$, $\frac{4}{7}$ ou $\frac{4}{8}$ où les 4 bits utiles sont codés respectivement par 5, 6, 7 ou 8 bits de transmission. Plus le taux de codage est faible ($\frac{4}{8}$), plus la fiabilité est grande, mais plus la durée de transmission est élevée. Le CR est réglé par défaut à $\frac{4}{5}$ [33].

3.2.4 Puissance de transmission (TP)

Comme la plupart des radios sans fil, les émetteurs-récepteurs LoRa permettent également d'ajuster la puissance de transmission (TP), en changeant radicalement la couverture radio et l'énergie nécessaire pour transmettre un paquet. En augmentant la TP, par exemple, de -4 à +20 dBm, la consommation d'énergie passe de 66 mW à 396 mW en utilisant l'émetteur-récepteur RFM95 [39]. Ainsi, une TP plus élevée augmente le rapport signal/bruit (SNR) au prix d'une augmentation de la consommation d'énergie de l'émetteur. Mais l'augmentation de la portée radio pourrait également entraîner une augmentation des interférences et du taux de collision.

3.3 Impact de la variation des paramètres sur la performance du réseau

Le choix des paramètres de transmission a un impact sur les performances de communication. Plus particulièrement, la sélection d'une combinaison des paramètres de transmission, appelée aussi une configuration, a un impact direct sur le débit et ToA, la portée de transmission et la résistance aux interférences, et sur la consommation d'énergie. Dans la plupart des situations, il est souhaitable d'équilibrer les performances de communication et la consommation d'énergie, car les nœuds sont alimentés par des batteries et l'objectif est de maximiser la durée de vie du réseau, tout en fournissant la QoS requise. Un autre facteur qui détermine la consommation d'énergie est le temps de transmission (ToA) nécessaire pour transmettre un paquet, qui dépend à son tour de la taille du paquet et le débit (DR) choisi. Table I.2 montre l'impact des différents paramètres de transmission sur les différents indicateurs de performances du réseau LoRaWAN.

Table I.2. Impact de la variation des paramètres sur la performance du LoRaWAN

	Débits	Énergie (ToA)	Portée (| RSSI |)	Interférences
SF ↗	-	+	+	+
BW ↗	+	-	-	-
TP ↗	NA	+	+	+
CR ↗	+	-	-	-

L'augmentation du SF augmente le RSSI pour une distance donnée entre le noeud et sa passerelle (gateway), ce qui traduit une augmentation de la portée puisque ce dernier en est proportionnellement dépendant. Cependant, il en résulte une augmentation du temps de transmission (ToA) et par conséquent la consommation énergétique. Étant donné que le SF comprend des valeurs de 7 à 12, une grande valeur augmente la sensibilité du récepteur au dépit d'un faible débit (DR). À l'inverse, une petite valeur du SF entraîne une augmentation du DR et une réduction de TOA (et par conséquent une diminution de la consommation d'énergie). A titre d'exemple, Table I.3 présente les DRs fournis par la spécification LoRaWAN dans [40]. Cependant, il existe environ 6720 combinaisons possibles référant à différents DR.

De plus, la distance par rapport à la passerelle a également une incidence sur RSSI. Plus le nœud est éloigné de la passerelle, plus le signal est faible à la réception, et donc plus le SF du noeud requis doit être élevé afin d'assurer que le RSSI reste au dessus du seuil pour décoder correctement le signal reçu.

En outre, dans les réseaux LoRaWAN, la puissance de transmission TP d'un paquet de données est réglable en fonction des besoins. En augmentant la TP, on consomme plus d'énergie, mais on élargie la portée du signal. Toutefois, une valeur TP élevée peut augmenter la probabilité de collisions, entraînant une augmentation d'interférences. Par ailleurs, le niveau de TP est limité réglementairement par des normes de CEM (Champs Électro-Magnétiques)

Enfin, le réglage d'une valeur CR faible ($\frac{4}{8}$), impliquant qu'il y a plus de bits de correction d'erreurs, assure une meilleure protection des données transmises contre les erreurs et les interférences, induisant une transmission plus lente et par conséquent plus d'énergie consommée.

Table I.3. Table des codes DR fournis par la spécification LoRaWAN [40]

Codes DR	Configuration	DR (bps)
0	SF12 - 125 kHz	250
1	SF11 - 125 kHz	440
2	SF10 - 125 kHz	980
3	SF9 - 125 kHz	1760
4	SF8 - 125 kHz	3125
5	SF7 - 125 kHz	5470
6	SF7 - 250 kHz	11000
DR8-DR15	RFU	Non défini

En résumé, l'amélioration d'un indicateur de performances en variant les paramètres de transmission est souvent en dépit de la dégradation d'un autre indicateur. Il est clairement souhaitable de choisir une configuration qui minimise la consommation d'énergie tout en répondant aux exigences de l'application en termes de performances du réseau. Toutefois, déterminer les paramètres qui réduisent le coût de l'énergie de transmission tout en maintenant les exigences de performance de communication est un défi. Il est vital que les nœuds alimentés par batterie sélectionnent les paramètres de transmission qui sont appropriés dans un réseau LoRaWAN. De mauvais choix pourraient entraîner un raccourcissement de la durée de vie des nœuds par un facteur de cent, rendant ainsi de nombreuses applications commerciales impossibles [34]. Il se pourrait aussi qu'un mauvais choix induise une perte de paquets importante face à des conditions instables. Des algorithmes permettant de trouver la configuration optimale des paramètres de transmission de chaque nœud sont nécessaires pour le réseau LoRaWAN. L'un des objectifs de notre thèse est de trouver une configuration optimale de ces paramètres assurant la fiabilité de transmission des applications avec une meilleure efficacité énergétique.

3.4 LoRaWAN

LoRaWAN correspond à la couche MAC du réseau LoRa. Contrairement à la couche physique de LoRa, elle est ouverte et entretenue par le groupe connu sous le nom de LoRa Alliance [20]. LoRa-Alliance décrit LoRaWAN comme suit : ≪ LoRaWAN™ est une spécification de réseau étendu à faible puissance (LPWAN) destinée aux noeuds sans fil fonctionnant sur batterie dans un environnement régional, national ou international, ou réseau mondial. LoRaWAN cible les exigences clés de l'Internet des objets, telles que la communication bidirectionnelle sécurisée, la mobilité et la localisation. La spécification LoRaWAN assure une interopérabilité transparente entre les objets intelligents sans nécessiter d'installations locales complexes et redonne la liberté à l'utilisateur, au développeur, aux entreprises facilitant le déploiement de l'Internet des objets ≫.

Comme le montre la citation ci-dessus, LoRaWAN a pour objectif principal d'être un simple protocole de réseau facile à déployer et répondant à toutes les exigences de base pour les noeuds IoT.

3.4.1 L'architecture du réseau LoRaWAN

La spécification LoRaWAN vise les noeuds sans fil fonctionnant sur batterie et permet de configurer facilement les noeuds souhaitant se connecter à un serveur de réseau. La Figure I.2 présente l'architecture du réseau LoRaWAN. Un réseau LoRaWAN a une topologie de réseau en étoile, se compose d'un serveur de réseau (racine du réseau), d'une ou plusieurs gateways et d'un ou plusieurs noeuds. Les noeuds peuvent être des capteurs ou d'autres entités produisant des données qu'ils souhaitent transmettre vers un serveur de réseau. Une gateway reçoit des données d'un ou de plusieurs noeuds qui lui sont connectés via LoRa et les transmet au serveur de réseau, agissant comme un relais transparent entre le noeud et le serveur de réseau. Un seul noeud peut également être connecté à plusieurs gateways. Le serveur de réseau met alors les données à la disposition d'un utilisateur ou d'une application. La communication entre un noeud et une gateway se fait par le protocole LoRa, tandis que la communication entre une gateway et un serveur de réseau peut se faire à travers IP, Ethernet, etc., ce qui signifie qu'une gateway doit être connectée à Internet d'une manière ou d'une autre.

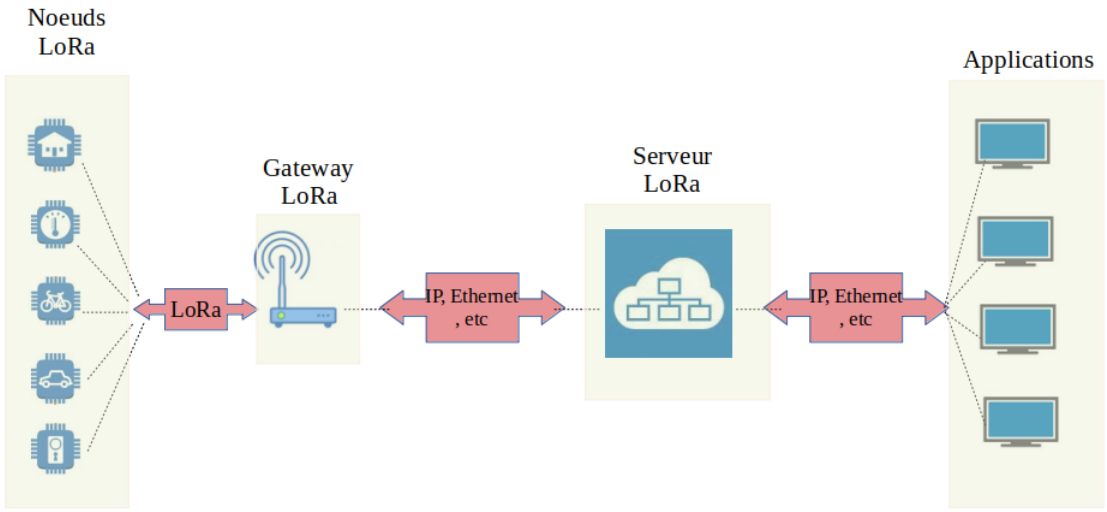


FIGURE I.2 – Architecture du réseau LoRaWAN

Afin d'augmenter l'efficacité spectrale, la durée de vie de la batterie et la portée, une gateway LoRaWAN peut négocier le débit de données, la puissance de sortie et les canaux de fréquence à utiliser avec les noeuds en utilisant un mécanisme d'adaptation de débit (ADR : Adaptive Data Rate) que nous discuterons plus loin. En outre, LoRaWAN prend en charge les émissions des gateways et les communications bi-directionnelles, mais avec des limitations. Ces limitations reflètent les cas d'utilisation des noeuds, ce qui donne trois classes de noeuds. Ces classes sont décrites dans la section suivante.

3.4.2 Classes des noeuds LoRa

LoRaWAN définit principalement trois types de classes de noeuds, en fonction des besoins du protocole, qui sont illustrés dans la Figure I.3. Il s'agit de classes A, B et C. Chaque classe assure un compromis entre la consommation d'énergie et la latence des communications.

La mise en œuvre des noeuds de classe A est obligatoire, tandis que les deux autres sont facultatives.

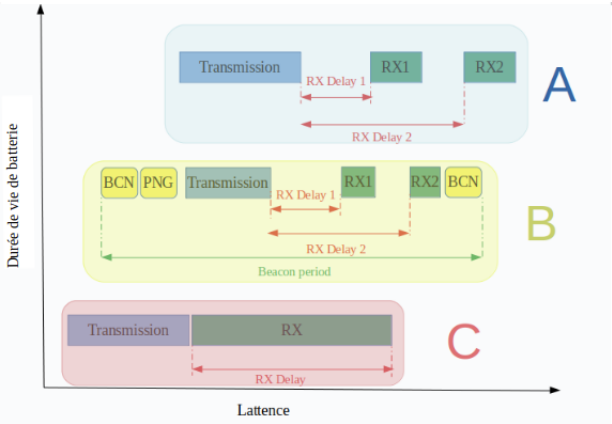


FIGURE I.3 – Classes des noeuds

3.4.2.1 Classe A (All)

Les noeuds de classe A ont les capacités de communication bidirectionnelle les plus limitées. Toutes les transmissions sur la liaison descendante (DL) vers un noeud de classe A doivent être effectuées après une transmission sur la liaison montante (UL) à partir de ce noeud. Cela est dû au fait qu'un noeud de classe A n'ouvre que deux courtes fenêtres de réception (RX1 et RX2) dans un délai déterminé à partir de sa transmission UL (Figure I.3). La transmission DL n'est pas possible en dehors de ces deux fenêtres de réception. Si une transmission DL est nécessaire à un autre moment, la gateway doit attendre la prochaine transmission UL à partir du noeud de classe A avant de transmettre son paquet DL. Selon la norme LoRaWAN [21], le serveur peut répondre à une transmission UL en envoyant une transmission DL en RX1, en utilisant le même canal et le même SF du paquet UL, ou en RX2, en utilisant un canal dédié (à 868,625 MHz en Europe) et SF 12 (c'est-à-dire le débit binaire le plus faible) pour maximiser la couverture. Si les paquets DL (l'acquittement ACK) sont reçus dans la première fenêtre de réception, la deuxième fenêtre de réception est ignorée. Comme mentionné précédemment, l'implémentation de la classe A est obligatoire, ce qui signifie que bien que la classe A puisse être modifiée pour fonctionner comme les classes B et C, le dispositif doit toujours être compatible avec les opérations de la classe A [21].

3.4.2.2 Classe B (Beacon)

Les noeuds de la classe B sont similaires aux noeuds de la classe A et doivent mettre en œuvre toutes les fonctionnalités des noeuds de classe A. En outre, les noeuds de la classe B permettent également plus de slots de réception en programmant l'ouverture des fenêtres de réception aux périodes prévues. Ces fenêtres de réception supplémentaires sont initiées après la réception d'un paquet de synchronisation de la gateway afin d'alerter le serveur que le noeud est en mode réception. Pour être bien synchronisé dans le temps et afin d'écouter sur un canal au bon moment les paquets DL, le serveur associe à chaque noeud de classe B une

gateway à travers laquelle il lui transmettra les beacons de synchronisation (BCN) [23]. Ces noeuds de classe B consomment plus d'énergie que ceux de classe A à cause de l'ouverture des fenêtres de réception supplémentaires.

3.4.2.3 Classe C (Continious)

Les dispositifs de classe C sont les mieux adaptés lorsque l'on s'attend à une transmission DL importante. Les dispositifs de classe C sont constamment à l'écoute des messages DL, c'est-à-dire que leur fenêtre de réception est toujours ouverte, sauf lors de la transmission de données UL. Les noeuds de classe C nécessitent une consommation d'énergie plus élevée que les deux autres classes, mais cela vient avec l'avantage d'une latence réduite (minimum) entre le nœud et le serveur [21].

Dans la transmission LoRa, il n'y a pas d'écoute du médium avant la transmission, la technique Pure ALOHA (P-ALOHA) est utilisée. Quand un noeud doit envoyer un paquet, ce dernier va être transmis immédiatement sur la ressource sans fil partagée entre tous les noeuds se trouvant dans le réseau et ceci en sélectionnant un canal pseudo-aléatoirement. Le noeud attend ensuite pendant un délai déterminé (RX1 et RX2) un acquittement du paquet envoyé par le destinataire de cette trame. Si le paquet est acquitté dans le délai prévu, on suppose que la transmission s'est effectuée correctement. Dans le cas contraire, le paquet aurait pu être victime d'une collision avec d'autres paquets envoyés sur le même canal avec le même SF ou que le noeud n'a pas envoyé le paquet avec la bonne configuration qui lui permet d'atteindre la gateway. Ce paquet a besoin d'être retransmis. Bien que cette technique d'accès soit très simple à implémenter, elle ne garantit pas la réception des paquets et peut engendrer des retransmissions augmentant par conséquent la consommation énergétique.

3.4.3 Limitation du Duty Cycle

Étant donné que la bande de fréquence ISM est utilisée par différentes technologies radio, le risque de collisions est fréquent. Pour limiter ce problème, les réglementations ETSI (Institut Européen des normes de Télécommunication) permettent le choix d'utiliser soit une limitation du cycle de service (duty cycle) correspondant au temps maximum pendant lequel un noeud puisse émettre, soit une gestion des transmissions basée sur l'écoute du canal dite Listen Before Talk Adaptive Frequency Agility (LBT AFA) similaire à CSMA du WiFi. La spécification LoRaWAN actuelle utilise exclusivement des transmissions limitées par duty cycle pour se conformer aux réglementations ETSI [40] [41].

En Europe, les noeuds fonctionnent dans la bande ISM ouverte de 868 MHz et doivent se conformer avec les règlements de l'ETSI pour la modulation à large bande. Cela permet aux noeuds LoRa de fonctionner sur des fréquences comprises entre 863 MHz et 870 MHz, mais avec des restrictions sur la durée de transmission. Les exigences en matière de Duty Cycle maximal pour l'accès au spectre sont très strictes et peuvent varier considérablement d'une bande à l'autre. Selon les paramètres régionaux européens pour LoRa, tous les noeuds doivent mettre en œuvre au moins les trois canaux de largeur de 125 KHz avec des fréquences centrales à 868,1 MHz, 868,3 MHz et 868,5 MHz. Cinq autres canaux supplémentaires existent dans la spécification LoRa. Ces canaux se limitent tous à un Duty Cycle de 1%, soit 36 sec/heure.

Face à des interférences et des pertes de paquets causées par une utilisation du même SF sur le même canal, par exemple, des retransmissions des paquets sont effectuées afin d'offrir plus de chance de réception. Cependant, un Duty Cycle à 1% pourrait réduire cette chance et entraînera plus de pertes à cause de l'interdiction de continuer à retransmettre au-delà de la limitation de duty cycle de 36 sec/heure.

3.4.4 Temps de transmission - Time On Air (ToA)

Le temps de transmission (ToA) est le temps écoulé depuis la transmission du premier bit d'information jusqu'au dernier [21]. Le ToA total est la somme du temps de transmission du préambule et le temps de transmission de la charge utile :

$$ToA_{trame} = T_{preamble} + T_{payload} \tag{I.4}$$

Les paramétres $T_{preamble}$ et $T_{payload}$ peuvent être calculés en utilisant la série de formules suivantes [40] :

$$T_{payload} = PL_{sym} \times T_{sym} \tag{I.5}$$

$$T_{sym} = \frac{2^{SF}}{BW} \tag{I.6}$$

$$T_{preamble} = (n_{preamble} + 4,25) \times T_{sym} \tag{I.7}$$

$$PL_{sym} = 8 + max\left(ceil\left(\frac{8PL - 4SF + 28 + 16 - 20H}{4(SF - 2DE)}\right)(CR + 4), 0\right) \tag{I.8}$$

La durée de transmission de la charge utile est alors la durée de transmission du symbole multipliée par le nombre de symboles de charge utile (PL_{sym}), défini dans l'Eq (I.5).

T_{sym} est le temps nécessaire pour envoyer 2^{SF} chips et est calculé à l'aide de l'Eq (I.6), où SF est le facteur d'étalement, BW est la largeur de bande.

L'équation (I.7) définit une durée de préambule où $n_{preamble}$ est la quantité de symboles du préambule (par défaut fixée à 8).

Le nombre de symboles qui composent la charge utile et l'en-tête du paquet est donné par l'Eq (I.8) où PL est la taille de la charge utile en octets, CR est le taux de codage utilisé (entre 1 et 4), H est l'option d'en-tête (0 si activé, 1 sinon) et DE est à 1 si l'optimisation du faible débit des données est activée sinon 0.

En excluant le facteur d'étalement, les valeurs de ces constantes ont été fixées par défaut et fournis par le firmware utilisé. La résolution de Eq (I.4) avec $n_{preamble} = 8$, PL = 10 (par exemple), CR = 1, H = 1, SF = 12, BW = 125 et DR = 0, résulte en un ToA total d'environ 990 ms, ce qui permet d'envoyer plusieurs paquets de 59 octets maximum (conformément aux paramètres régionaux LoRaWAN) en un cycle de 36 secondes [40].

3.4.5 Procédures de jointure (ABP et OTAA) et Sécurité

Lorsqu'un nouveau noeud est ajouté à un réseau LoRa, il doit passer par un processus d'association appelé « procédure de jointure ». Lors de ce processus, deux clés de sécurité sont partagées entre le noeud et le serveur. Actuellement, LoRa propose deux types de méthodes d'activation présentées dans la Figure I.13).

— Over-The-Air Activation (OTAA)

— Activation By Personnalisation (ABP)

3.4.5.1 Procédure OTAA [42]

Pour OTAA, les noeuds doivent suivre une procédure de jointure avant de participer aux échanges de données avec le serveur du réseau. Cette procédure offre un moyen flexible et sûr pour définir des clés de session avec les serveurs et constitue la procédure d'activation recommandée.Elle exige que le noeud soit personnalisé avec des informations uniques ($DevEUI$ et $AppEUI$). A travers ces informations personnalisées pour chaque noeud, une négociation avec le serveur est effectuée. Celle-ci consiste en un échange de deux messages MAC, à savoir une demande de jointure (Join Request) et une acceptation de jointure (Join Accept). Les noeuds transmettent le Join Request qui sera traité par le serveur, qui le valide et répond avec un message Join Accept. Le Join Request est transmis avec les paramètres de transmission permettant la plus grande portée afin de garantir la réception du message par les gateways. En utilisant la clé racine AES-128 spécifique au noeud (App_Key, configurée avant le déploiement) et les informations contenues dans le message Join Accept, les noeuds dérivent leurs clés de session (Nwk_SKey et App_SKey).

3.4.5.2 Procédure ABP [42]

Dans certaines circonstances, les noeuds peuvent être activés par personnalisation (ABP). L'activation par personnalisation est la méthode la plus simple, mais la moins sécurisée. Elle lie directement un noeud à un réseau spécifique, en contournant la procédure de demande de jointure. Ainsi, contrairement à OTAA, le noeud est déjà équipé des informations nécessaires pour participer à un réseau LoRa spécifique. Le DevAddr et les deux clés de session Nwk_SKey et App_SKey sont déjà stockées dans le noeud au lieu du DevEUI, de l'AppEUI et de l'App_Key.

L'avantage de l'ABP est la facilité de se connecter au réseau, car le noeud peut être rendu opérationnel en peu de temps, ce qui est bien adapté à certaines applications n'exigeant peu de sécurité. L'inconvénient est que les clés de cryptage (Nwk_SKey et App_SKey) permettant de communiquer avec le réseau sont pré-configurées dans le noeud, ce qui affaiblit la sécurité.

3.4.6 Format du message

LoRa et LoRaWAN définissent différents formats de messages. Dans cette section, nous présentons le format du message MAC en spécifiant les messages d'enregistrement des noeuds (Join Request et Join Accept).

3.4.6.1 Format du message MAC

Les messages MAC de LoRaWAN sont contenus dans la charge utile $PHYPayload$ de la couche Physique de LoRa. La structure de $PHYPayload$ est illustrée dans la Figure I.4.

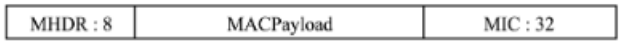


FIGURE I.4 – Structure de PHYPayload

En outre, le champ ($MACPayload$) peut correspondre à une demande de jointure de réseau (Join Request) ou une réponse de jointure (Join Accept), si nécessaire. L'en-tête MAC ($MHDR$) et le contrôle d'intégrité du message (MIC) sont fixés à une longueur de 8 bits et 32 bits respectivement. La charge utile MACPayload (Voir Table I.4 est cependant de taille dynamique avec une longueur maximale variable selon les configurations utilisées présentées dans la Table I.3.

FIGURE I.5 – Structure de MACPayload

Table I.4. Table des codes DR fournis par la spécification LoRaWAN [40]

Codes DR	Configuration	Taille MACPayload
DR0	SF12 - 125 kHz	59
DR1	SF11 - 125 kHz	59
DR2	SF10 - 125 kHz	59
DR3	SF9 - 125 kHz	123
DR4	SF8 - 125 kHz	230
DR5	SF7 - 125 kHz	230
DR6	SF7 - 250 kHz	230
DR8-DR15	RFU	Non défini

La structure de MACPayload est illustrée dans la Figure I.5 et contient l'en-tête du paquet $FHDR$, un port de trame ($FPort$) et une charge utile de trame ($FRMPayload$) [21].

De plus, le $FHDR$ de la charge utile MAC contient quatre champs qui sont utilisés par le protocole LoRaWAN. Comme le montre la Figure I.6, ces champs sont l'adresse du dispositif ($DevAddr$ sur 32 bits), le contrôle de trame ($FCtrl$ sur 8 bits), le compteur de trame ($FCnt$ sur 16 bits) et les options de trame ($FOpts$ sur 120 bits). Au total, le $FHDR$ est long de 7 à 22 Bytes tout dépend de si les options de ($FOpts$) sont utilisées ou non.

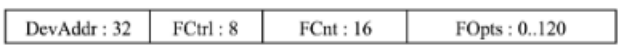


FIGURE I.6 – Structure de FHDR

Les Figures I.7, I.8, I.9, et I.10 détaillent respectivement les champs : $MHDR$, $FCtrl$, et $FOpts$ qui sont exploités plus loin pour implémenter nos propositions futures. Le champ

MHDR dans la Figure I.7 contient un champ pour spécifier le type de message envoyé (*MType* sur 3 bits), un champ *Major* pour indiquer la version LoRaWAN utilisée (sur 2 bits), et un champ libre (*RFU*) sur 3 bits pour les utilisations futures que nous exploitons plus loin pour l'intégration de nos nouvelles fonctionnalités.

Le champ *FOpts* est utilisé pour transporter les commandes MAC dans LoRaWAN [21]. Ce champ commence par un identifiant de commande (par exemple CID=$0x0B$) et ensuite le contenu des options a intégrer [21].

FIGURE I.7 – Structure de MHDR

Bit#	7	6	5	4	[3..0]
FCtrl bits	ADR	ADRACKReq	ACK	ClassB	FOptsLen

FIGURE I.8 – Structure de FCtrl UL

Bit#	7	6	5	4	[3..0]
FCtrl bits	ADR	RFU	ACK	FPending	FOptsLen

FIGURE I.9 – Structure de FCtrl DL

FIGURE I.10 – Structure de FOpts

3.4.6.2 Format des messages de jointure

Comme mentionné précédemment, dans le cas d'une activation OTAA, deux messages sont échangés entre les noeuds et le serveur afin de confirmer ou refuser la procédure de jointure : le message de demande de jointure (Join Request) et le message de réponse à la demande de jointure (Join Accept). Dans ce qui suit, nous présentons le format et les différents champs contenus dans ces deux messages.

— **Join Request :**
Le noeud commence la procédure de jointure en envoyant un Join Request. La Figure I.11 illustre le format du message Join Request. Ce dernier inclut des champs contenant des informations uniques à chaque noeud : *DevEUI* (8 Bytes), *AppEUI* (8 Bytes), et *DevNonce* (2 Bytes). *DevEUI* et *AppEUI* réfèrent respectivement aux identifiants globales du noeud et de l'application. *DevNonce* est un compteur qui commence par 0 lorsque le noeud est initialement activé et est incrémenté après chaque transmission d'un Join Request par un noeud. La valeur du *DevNonce* ne doit jamais être réutilisée pour une même valeur *AppEUI*. La ré-initialisation de *DevNonce* sans modifier l'*AppEUI*

entraînera le serveur à rejeter les demandes de jointure du noeud. Pour chaque noeud, le serveur conserve la trace de la dernière valeur $DevNonce$ utilisée par le noeud, et ignore les demandes de jointure si DevNonce n'est pas incrémenté.

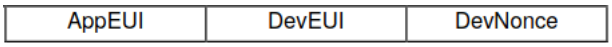


FIGURE I.11 – Format du Join Request

— **Join Accept :** Le serveur répondra au message Join Request par un message Join Accept si le noeud est autorisé à participer à un réseau. Le message Join Accept contient un $AppNonce$ de 3 Bytes, un identifiant de réseau $NetID$ de 3 Bytes, une adresse du noeud $DevAddr$, un délai $RxDelay$ (1 Byte) entre la transmission (TX) et la réception (RX) correspondant au temps d'attente avant ouverture de la première fenêtre de réception ($RX1$) et une liste optionnelle de fréquences de canaux $CFList$ (16 Bytes) qui peuvent être utilisées par le noeud. Le $DevAddr$ est un identifiant 32 bits du noeud au sein du réseau actuel. Les 7 MSB du $DevAddr$ sont appelés $NwkID$, qui est également contenu dans $NetID$. Les autres bits peuvent être choisis arbitrairement par le serveur du réseau. Enfin, l'ensemble du message d'acceptation de la jointure est crypté avec l'App_Key.

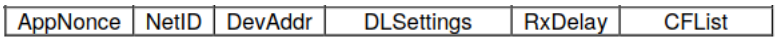


FIGURE I.12 – Format du Join Accept

3.4.7 Sécurité dans LoRaWAN

Pour des raisons de sécurité, la procédure de jointure OTAA est celle utilisée fréquemment dans le réseau LoRaWAN où des clés de cryptage sont générées par le serveur au lieu de les pré-configurer directement dans le noeud comme le propose la procédure ABP.

Lorsque les noeuds envoient leurs messages de demande de jointure « Join Request », un code d'intégrité du message - Message integrity code (MIC) est calculé en utilisant Eq (I.9) et Eq (I.10).

$$cmac = aes128\_cmac(AppKey, JoinRequest). \tag{I.9}$$

$$MIC = cmac[0...3]. \tag{I.10}$$

Lorsque le serveur reçoit le « Join Request », il vérifie si le paramétre $DevNonce$ ne correspond pas à une valeur $DevNonce$ précédente. Si la valeur existe déjà, le serveur rejette le message. Si le message est accepté, le serveur authentifie le noeud en calculant le MIC et en vérifiant si ce dernier correspond à celui calculé par le noeud afin de décider de continuer la génération des clés de session ou non. Les clés de session d'application sont générées en utilisant Eq (I.11) et Eq (I.12) :

$$Nwk\_SKey = aes128\_encrypt(AppKey, 0x01|AppNonce|NetID|DevNonce|pad\_16). \tag{I.11}$$

$$App\_SKey = aes128\_encrypt(AppKey, 0x02|AppNonce|NetID|DevNonce|pad_{16}). \tag{I.12}$$

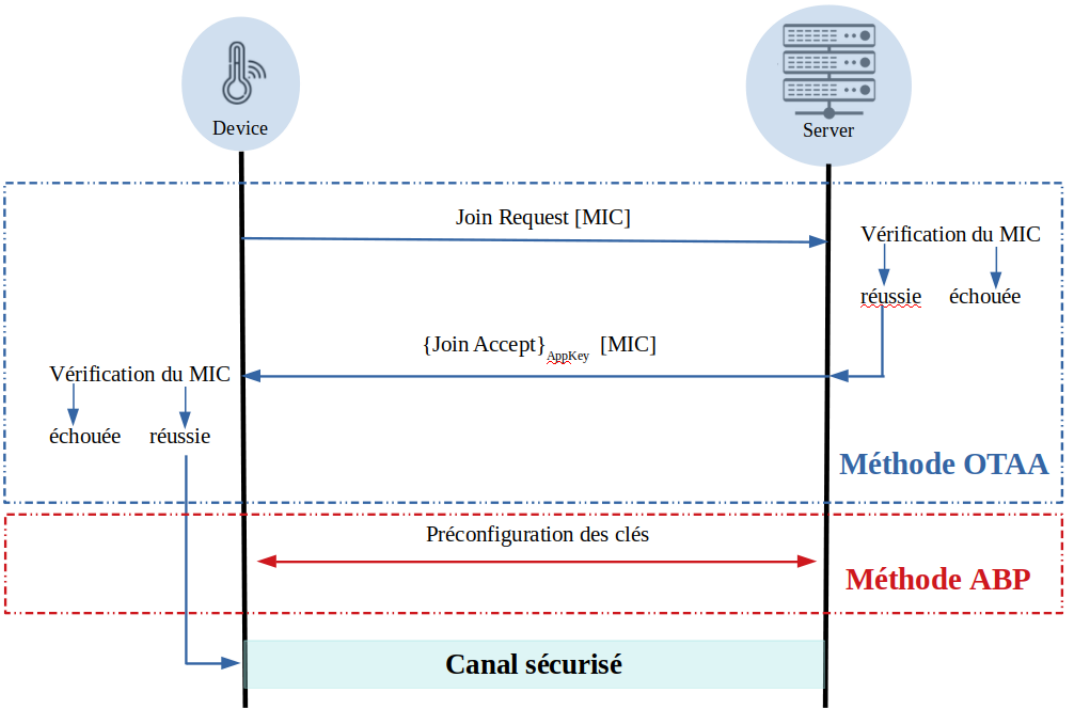


FIGURE I.13 – Procédures de jointure et protocole de sécurité LoRaWAN

Ensuite, le serveur envoie son « Join Accept » crypté par la clé *App_Key* contenant un paramètre AppNonce (initialement fixé à 0, puis incrémenté à chaque envoi du message « Join Accept » au noeud correspondant) afin de décrypter le message et de générer les clés de session *App_SKey* et *Nwk_SKey*.

Dans LoRaWAN, les noeuds utilisent l'algorithme AES 128 pour crypter les messages en utilisant le mode « Counter » (CTR) [21]. Le cryptage AES implémenté dans LoRaWAN n'est utilisé que pour générer une série de clés, et non pas pour crypter les messages. Cela signifie que les octets du bloc de messages ne sont pas mixés comme dans un vrai cryptage AES. Cette série de clés est XORée avec les données comme dans Eq (I.13), Eq (I.14) et Eq (I.15). Ces équations expriment les processus de cryptage de la charge utile du paquet dans le LoRaWAN en utilisant un cryptogramme où S_i indique le bloc compteur et K est la clé de cryptage. Le résultat du cryptage n'est pas optimal et pourrait être déchiffré par un agresseur, car, d'une part, l'opération XOR donne lieu à des messages cryptés de la même longueur que la clé. D'autre part, le format du message crypté reste le même et sous une forme connue par n'importe quel utilisateur puisqu'il n'y a pas eu de remaniement d'octets comme dans un vrai

cryptage AES, ce qui n'empêcherait pas des attaques sur des messages chiffrés provoquant un changement dans la valeur du MIC et donc un rejet de paquet.

$$S_i = aes128\_encrypt(K, A_i), \quad for \quad i = 1...k \tag{I.13}$$

$$S = S_1|S_2|...|S_k \tag{I.14}$$

$$(pld|pad_1 6)\ XOR\ S \tag{I.15}$$

De plus, l'envoi du « Join Request » contenant des informations personnalisées est en clair et à disposition des noeuds malveillants. Cela laisse des risques d'attaques de type usurpation d'identité (ID-Spoofing), par exemple, que nous allons étudier également dans cette thèse.

3.4.8 Procédure d'adaptation de débit (ADR)

Afin de parvenir à un mode de fonctionnement optimal en termes de performance réseau (fiabilité, portée, délai, débit et consommation d'énergie), LoRaWAN offre la possibilité d'utiliser différents débits en manipulant les paramètres mentionnés précédemment (SF, BW, CR et TP) selon le besoin de l'application. [34] a défini 6720 combinaisons de ces paramètres donnant lieu à des performances différentes. Pour répondre aux besoins de performances du réseau d'une part, et pour minimiser la consommation d'énergie (et donc maximiser la durée de vie des batteries des nœuds) d'autre part, LoRaWAN définit un mécanisme d'adaptation de débit - Adaptive data rate (ADR) pour gérer individuellement le débit en choisissant une configuration permettant d'augmenter le débit (faible SF qui minimise le ToA et donc la consommation d'énergie) pour chaque nœud. L'utilisation de l'ADR a pour objectif d'augmenter la capacité d'un réseau LoRaWAN, puisque les paquets de données qui sont transmis à l'aide de différents SFs sont orthogonaux et peuvent être transmis simultanément [8]. L'ADR contrôle les paramètres de transmission des données UL du nœud à la passerelle. L'algorithme ADR est responsable de la gestion du débit de données (DR) et de la puissance de transmission (TP) des nœuds sur la base de l'estimation du budget de liaison TOA dans le message UL et du SNR minimum requis pour décoder avec précision les paquets de données au débit de données existant. ADR est proposé pour des noeuds fixes, où le SNR est calculé en fonction de l'historique des paquets UL reçus, appelé "ADR géré par le réseau, ADR statique, ou ADR-Server". L'approche "ADR basée sur le réseau" ne fonctionne pas pour les nœuds mobiles en raison de l'atténuation du signal rapide qui se produit lorsque le noeud se déplace. Quand le réseau est incapable de contrôler le débit du noeud, c'est la couche application de ce dernier qui devrait le contrôler. Dans ce cas, l'ADR s'effectue "en aveugle" du côté du nœud terminal appelé "ADR Blind ou ADR-Node" [43].

Une fois l'ADR activé (bit ADR positionné) le serveur utilise une commande MAC *LinkADRReq* pour contrôler le débit de données et la TP du nœud. Le nœud répondra par la commande *LinkADRAns* pour indiquer l'acceptation ou le rejet des nouveaux paramètres. Le bit ADR peut être activé ou désactivé par le noeud ou par le serveur à la demande. Une fois son débit est augmenté par le serveur, le noeud a besoin périodiquement de vérifier que ses paquets arrivent bien au serveur en exigeant un acquittement (ACK) pour chaque paquet envoyé. Si un message ACK n'est pas reçu par le noeud au bout d'un certain temps, celui-ci

passe à un débit de donnée inférieur pour tenter de rétablir la connectivité, mais cela après plusieurs tentatives de retransmission. Bien que LoRaWAN stipule un schéma de signalisation des paramètres de transmission via la commande *LinkADRReq*, il n'existe aucune description de la manière dont la communication doit être traitée. La spécification n'indique pas comment le serveur doit donner des instructions aux nœuds concernant l'adaptation du débit de données, le moment où il faut modifier un paramètre ou l'ordre dans lequel les paramètres doivent être modifiés [44].

Initialement, lors de la phase de procédure de jointure à un réseau LoRaWAN, le nœud utilise souvent SF12 (plus grande portée radio) pour s'assurer d'atteindre une gateway (ou une valeur de SF de manière aléatoire, selon le fabricant du nœud). Après la réception de 20 paquets successifs, applique ADR-Server (Algorithme 1) pour choisir un SF approprié (plus petit que le SF initial utilisé par le nœud) en fonction des valeurs SNRs (Signal sur bruit) pour essayer d'optimiser le débit de données, le ToA et par conséquent de réduire la consommation d'énergie. Dans ce qui suit nous présentons les deux algorithmes ADR (ADR-Node et ADR-Server) [3].

Dans l'Algorithme 1, le serveur réseau estime la marge SNR (SNR_{margin}) sachant que SNR_{max} est le maximum des SNR des 20 derniers paquets reçus et le SNR requis (SNR_{Req}) est une valeur par défaut pour chaque combinaison de configuration (CIR). Cette marge de SNR sera utilisée pour estimer dans quelle mesure nous pouvons augmenter le débit (en diminuant le SF ou incrémentant la TP).

Après le calcul de SNR_{margin}, un nombre d'étapes N_{step} est calculé pour déterminer le nombre de fois que nous allons diminuer le SF ou incrémenter la TP.

— Dans le cas où $N_{step} > 0$ et le SF minimum n'est pas encore atteint, le serveur diminue SF (augmente DR). Lorsque le plus petit SF est déjà atteint, le serveur songe à diminuer la TP jusqu'à ce que sa valeur atteigne la valeur minimale (TX_{min}=2 dBm).

— Dans le cas $N_{step} < 0$ et puissance maximale de transmission n'est pas atteinte, le serveur augmente la puissance de transmission jusqu'à ce que la puissance maximale transmise soit atteinte (TX_{max}= 14 dBm en Europe).

L'objectif de ADR-Node est uniquement d'augmenter la portée en augmentant le SF si les données n'atteignent pas la gateway (transmissions perdues). L'algorithme 2 présente l'algorithme du ADR-Node. Un compteur $ADRACKCNT$ définissant le nombre de retransmissions d'un paquet n'ayant pas été acquitté est déclenché pour chaque transmission UL. Si ce compteur atteint une limite fixée $ADRACKLIMIT$(= 64 pour l'EU 868) sans recevoir d'acquittement (transmission DL), le nœud doit augmenter le SF pour une plus grande portée. Ce compteur est réinitialisé lors de la réception d'acquittement.

À l'heure actuelle, le mécanisme ADR proposé par Semtech ne prend en charge que les noeuds LoRa fixes et n'est pas défini pour le cas de mobilité bien que certains travaux l'ont testé dans des scénarios de véhicules connectés [45–47]. Cependant, ils ont noté un taux de perte de paquets élevé. Dans cette thèse, nous allons proposer des nouveaux algorithmes permettant d'étendre ADR afin de renforcer les performances de LoRaWAN et de lui permettre de supporter des noeuds mobiles.

Algorithme 1 : ADR-Server

input : DataRate, TP, SNR_{max}, SNR_{Req}
output : DR, TP

$SNR_{margin} \leftarrow$ SNR$_{max}$ - SNR_{Req} - 10
$N_{step} \leftarrow floor$(SNR$_{margin}$/3)
DR $\leftarrow getDataRateIndex(DataRate)$
for $N_{step} > 0$ *et* $DR < maxDR$ **do**
  DR $\leftarrow DR + 1$&N$_{step}$ $\leftarrow$N$_{step}$-1
end
for $N_{step} > 0$ *et* $TP > minTP$ **do**
  TP $\leftarrow TP - 3$&N$_{step}$ $\leftarrow$N$_{step}$-1
end
for $N_{step} < 0$ *et* $TP < maxTP$ **do**
  TP $\leftarrow TP + 3$&N$_{step}$ $\leftarrow$N$_{step}$-1
end

Algorithme 2 : Algorithme ADR-Node

if *Transmission UL* **then**
  ADRACKCNT $\leftarrow ADRACKCNT + 1$
  if $ADRACKCNT == ADRACKLIMIT$ **then**
    SF $\leftarrow SF + 1$
    if *Transmission DL reçue* **then**
      ADRACKCNT $\leftarrow 0$

4 – Conclusion

Au sein de ce chapitre, nous avons présenté les différentes technologies de communications sans fil utilisées dans le cadre de l'internet des objets. Nous nous sommes focalisés principalement sur leurs caractéristiques de performances en termes de débit, portée, autonomie, utilisation des bandes ISM sans licence et le coût. À travers la première partie de ce chapitre, nous avons discuté les avantages des réseaux LPWANs pour s'accommoder aux exigences de l'IoT. Nous avons ensuite comparé les différentes technologies LPWAN en discutant les avantages et les inconvénients de chacune. Dans la deuxième partie du chapitre, nous nous sommes focalisés principalement sur la technologie LoRa qui offre plus d'avantages que les autres réseaux LPWAN. Toutefois, malgré les réponses aux exigences IoT, ces LPWANs permettent d'obtenir une QoS limitée. Pour cela, la QoS du réseau LoRa est l'un des principaux défis futurs, ce qui a fait l'objet d'études dans notre thèse. En effet, nous concluons que le protocole LoRaWAN est confronté à plusieurs limitations influençant la QoS du réseau, telles qu'une restriction de Duty Cycle, possibilité de collisions, mauvaise adaptation aux environnements variables (mobilité, par exemple), etc. Nos prochains chapitres présenteront nos contributions proposant des solutions pour la gestion de QoS pour une meilleure performance du réseau LoRaWAN dans des environnements hétérogènes, utilisées pour plusieurs domaines d'applications (À titre d'exemple, un scénario de ferme connectée et un scénario d'école connectée faisant apparaître les besoins de mobilité et de transmission des données de volumes très hétérogènes).

Chapitre II

Extension d'ADR pour le support de la mobilité

Dans ce chapitre, nous présentons les défis concernant le mécanisme d'adaptation de débit - Adaptive Data Rate (ADR) proposé par LoRaWAN ainsi que les différentes variantes de ce mécanisme présentées dans la littérature. Nous étudions les limitations de ces versions dans un environnement mobile, et dans ce contexte nous proposons une extension d'ADR baptisée « E-ADR ». Ce chapitre porte sur la proposition de deux versions du mécanisme proactif E-ADR basées sur des solutions d'apprentissage et de prédiction. La première version tient compte des mobilités à trajectoires connues. La deuxième version est une extension permettant d'élargir la liste d'applications IoT utilisant le protocole LoRaWAN dans des contextes de mobilité inconnue.

1 – Introduction

Afin d'augmenter les performances du réseau en termes de débit (via principalement la diminution de SF, qui par conséquent diminue le TOA et la consommation d'énergie, cf. Table I.2, ligne 1), LoRaWAN propose un mécanisme d'adaptation de débit (ADR), appelé aussi l'ADR basique dans la suite. L'objectif de ce schéma est d'adapter et d'optimiser le débit (DR) et la puissance de transmission (TP) des noeuds statiques [21] afin de leur offrir le débit le plus élevé possible. Toutefois, le choix du débit est un compromis entre la portée de la communication et le temps de transmission des messages (ToA) [21]. Rappelons que l'ADR contrôle les paramètres de transmission, à savoir la largeur de bande (BW), le facteur d'étalement (SF), la puissance de transmission (TP) et le taux de codage (CR). D'une part, l'optimisation du débit augmente la capacité du réseau, diminue le TOA et la consommation d'énergie, d'autre part, elle conduit à l'adoption d'un faible SF, qui à son tour diminue la porté radio effective car réduit la sensibilité de la réception (seuil RSSI détectable plus élevé [23]). Vouloir trop "optimiser" le débit risque de rendre la transmission plus sensible à des perturbations. Notons que par ailleurs, les paquets de données transmis avec différents SFs sont orthogonaux et peuvent donc être reçus simultanément, ce qui offre une opportunité de réduire le temps de transmission global d'un ensemble de noeuds en adoptant chacun un SF (et donc DR) différent, qui n'est pas forcément optimal individuellement.

L'ADR est un mécanisme qui contrôle les paramètres de transmission de la liaison montante (UL) des noeuds LoRa en fonction du budget de la liaison. Cependant, la spécification

LoRaWAN [21] n'a pas défini un algorithme précis pour que le serveur informe les noeuds de la meilleure configuration à adopter pour la prochaine émission. Il en résulte une lacune en termes de mise en œuvre de l'ADR, ce qui explique que de nombreux schémas ADR basés sur différents modèles mathématiques, des simulations et des bancs d'essai ont été proposés. En outre, comme ADR est plutôt adapté aux noeuds statiques, comme ce que nous allons montrer, il ne peut pas être efficace pour une grande catégorie d'applications IoT (par exemple, suivi des animaux, tracteurs et drones connectés dans une ferme, Bus scolaires connectés, etc.) qui implique non seulement des noeuds statiques, mais également des noeuds mobiles [48] où une reconfiguration rapide est nécessaire afin de réadapter de nouveau des paramètres de transmission rendant la configuration optimale.

Dans ce contexte, un grand défi est de déterminer en temps réel la bonne configuration qui permet une communication fiable avec une faible consommation d'énergie, dans des environnements instables. En effet, l'adaptation du débit actuel n'est effectuée qu'après la réception de plusieurs paquets (20 paquets dans l'ADR Basique) [3] ce qui résulte en une lente adaptation aussi bien dans le cas de noeuds fixes en présence d'obstacles temporaires que dans le cas de noeuds mobiles lorsque l'atténuation du canal radio change fréquemment.

Plusieurs travaux ont essayé d'évaluer ADR Basique [45–47] sans proposer d'améliorations. Nous avons à notre tour effectué quelques tests en utilisant des cartes LoRa Waspmote de libelium [49]. Nous nous sommes basés dans ces tests aux différentes configurations (modes de 1 à 10) présentées dans Table II.1. Ces tests ont révélé quatre problèmes :

— Tout d'abord, le réglage selon le meilleur SNR suppose que le réseau fonctionne dans des conditions stables. Cependant, les conditions climatiques, les interférences, les obstacles en mouvement et les nœuds mobiles peuvent modifier radicalement le SNR réel. Ainsi, le fait de se baser uniquement sur la valeur maximale des 20 derniers paquets reçus ne conduit pas nécessairement au choix optimal.

— Deuxièmement, si on prend l'exemple où le réseau reçoit 20 paquets utilisant une configuration *mode*2 (Voir Table. II.1) et un SNR maximal de SNR_{max}= 5 dB. Selon l'Algorithme 1, la marge SNR est égale à SNR_{margin}=14 dB. Ceci se traduit en une sur-utilisation de paramètres de configuration, donc le serveur devrait augmenter le débit afin de réduire la consommation énergétique. Pour cela, il calcule le nombre d'étapes pour la réduction du SF en utilisant l'Algorithme 1 qui résultera en $N_{step} = 4$. Le serveur attribue donc *mode*6 au noeud. En se référent à la Table. II.1, le SNR par défaut du *mode*6 est de -15 dB alors que SNR_{max}= 5 dB. Nous concluons que bien que l'ADR ait réduit le temps de transmission ainsi que la consommation énergétique en augmentant le débit, la marge SNR reste encore trop élevée en utilisant *mode*6 (SNR_{margin}=10 dB). De plus, en se référant à Table. II.1, le mode le plus optimal serait *mode*10 correspondant à un SNR de -7,5 dB. Ce dernier se rapproche le plus du SNR_{max} reçu. Donc l'algorithme ADR ne propose pas un réglage suffisamment fin des paramètres de configuration ce qui l'empêche d'atteindre une allocation optimale répondant aux besoins d'un noeud.

— Troisièmement, attendre de recevoir 20 paquets avant la reconfiguration peut prendre

trop de temps, ce qui empêche l'adaptation rapide aux changements rapides des conditions environnementales (par exemple, la mobilité des nœuds, ou présence d'obstacles temporaires dans le cas des noeuds fixes).

— Et enfin, aucune attention n'a été accordée à la limite ToA dans le cadre de la restriction du Duty Cycle du noeud (1%). Si le noeud est attribué une configuration avec un grand SF, il risque d'atteindre rapidement la limite du Duty Cycle. D'autre part, une configuration non-adéquate à la position du noeud, entraînera à un taux de perte de paquets plus élevé induisant un grand nombre de retransmissions qui entraînera une importante consommation du ToA. C'est pour cela qu'un réglage plus fin est recommandé pour une QoS plus optimale et plus performante.

Table II.1. les modes de configuration pour TP=14dB [49]

mode	BW (kHz)	CR	SF	RSSI(dB) [Bi , Bs]	DR (Kbps)	SNR-Req (dB)
1	125	4/5	12	$[-134, -131]$	0,293	-20
2	250	4/5	12	[-131 , -129]	0,585	-19
3	125	4/5	10	[-129 , -128]	0,976	-18.2
4	500	4/5	12	$[-128, -126]$	1,718	-17.5
5	250	4/5	10	$[-126, -125.5]$	1,953	-16
6	500	4/5	11	$[-125.5, -123]$	2,148	-15
7	250	4/5	9	$[-123, -120]$	3,515	-12.5
8	500	4/5	9	$[-120, -117]$	7,031	-11
9	500	4/5	8	$[-117, -114]$	12,50	-10
10	500	4/5	7	$[\geq -114]$	21,875	-7.5

Plusieurs schémas d'adaptation du débit de données ont été proposés dans la littérature qui tentent d'améliorer les performances de communication dans un réseau LoRaWAN [4, 5, 34, 50–52], mais aucun n'a traité explicitement la mobilité des nœuds ni la contrainte du Duty Cycle.

Pour mieux gérer la mobilité des nœuds ainsi que les obstacles et réduire la probabilité d'atteindre le Duty Cycle de 1%, nous proposons E-ADR (Enhanced ADR) un mécanisme d'adaptation de débit amélioré visant à auto-adapter la configuration en présence d'obstacles temporaires devant des noeuds fixes ou en cas de noeuds mobiles grâce à une reconfiguration rapide des paramètres de transmission LoRaWAN. E-ADR est basé sur la prédiction des valeurs RSSIs (Indicateur de force du signal reçu) des prochains paquets selon la variation des anciennes valeurs en utilisant la méthode de régression linéaire. L'idée principale est de réajuster le SF pro-activement afin d'éviter la perte de paquets causée par une mauvaise configuration (mauvais choix de SF). E-ADR ne se limite pas seulement à la minimisation du SF pour un meilleur débit et un minimum de ToA, comme le cas du ADR Basique et les différentes variantes de la littérature, mais il peut aussi incrémenter le SF pour s'adapter rapidement à un éloignement du noeud de la gateway.

Dans la suite, nous donnons d'abord un état de l'art sur les différentes propositions d'amélioration d'ADR de la littérature, puis nous présentons les concepts de notre proposition

d'extension E-ADR qui permet d'étendre LoRaWAN pour le support des noeuds mobiles.

2 – Les variantes ADR dans la littérature

Dans la littérature, le schéma ADR a été modifié et mis en œuvre pour répondre à différents objectifs du LoRaWAN en ciblant les mesures de performance du réseau. Dans cette section, nous présentons plusieurs axes de recherches et études pertinentes liés aux performances et à la QoS du protocole LoRaWAN. Certaines recherches ont porté sur l'évaluation de performances du protocole LoRaWAN en termes de scalabilité, de durée de vie de la batterie et de couverture (RSSI) [53–57]. D'autres travaux de recherche se sont intéressés à l'évaluation de performances de LoRa pour des applications mobiles [45–47] par le biais de mesures expérimentales, sans proposer de solutions, en particulier face au problème de taux de perte de paquets élevé. Certains travaux [4, 5, 34, 50–52] ont visé à améliorer les performances du protocole LoRaWAN, en apportant des améliorations à l'ADR ou des améliorations du débit des transmissions UL [58].

2.1 Évaluation de performances de ADR dans un contexte de mobilité

Les auteurs de [45] se sont concentrés sur l'impact de la mobilité sur la performance des transmissions de LoRa, à travers un scénario de véhicule pour la collecte de données à longue distance. Ils ont montré que la mobilité affecte les communications LoRa en réduisant le taux de réception des paquets (PRR). Les auteurs dans [46] se sont intéressés à l'étude de la pertinence de LoRa dans un contexte de mobilité. À cette fin, ils ont étudié les performances de l'ADR dans divers scénarios de mobilité pour des véhicules connectés en faisant varier la vitesse de déplacement. Ils ont confirmé par des expérimentations que, bien que l'ADR implémenté par Semtech améliore la fiabilité et la couverture du réseau mobile à très faible mobilité, son PRR diminue au fur et à mesure que la mobilité augmente, laissant ainsi de la place pour des améliorations et des optimisations ultérieures dans leurs travaux futurs. D'autres tests dans le contexte de la communication véhiculaire ont été menés dans [47]. Les auteurs ont évalué l'impact des obstacles entre les nœuds fixes et la gateway LoRa en termes de PRR. En outre, les nœuds ont été placés sur des véhicules et chaque véhicule a suivi une trajectoire différente dans les rues proches du bâtiment où se trouvait la gateway. Le chemin à parcourir pour chaque véhicule était d'environ 2 km. L'expérimentation dans ce scénario mobile a montré un taux de perte de paquets élevé (plus de 60%). Tous ces travaux confirment l'intérêt d'envisager des améliorations du principe ADR afin de réduire le taux de perte (PLR).

2.2 Adaptation de la Configuration des transmissions descendantes

Contrairement aux autres propositions qui concernent l'adaptation de débit des transmissions montantes (ADR), les auteurs dans [58] visent à améliorer le débit et trouver le bon paramétrage pour les transmissions descendantes. Les auteurs se sont concentrés sur l'étude

d'un scénario visant les grands réseaux à trafic bi-directionnel. Ce scénario permet d'observer certains effets imprévus découlant de l'interaction de multiples noeuds desservis par une seule gateway et un serveur. Le but de leur étude est d'obtenir une compréhension plus approfondie du rôle joué par chaque élément configurable sélectionné par les auteurs, d'identifier les comportements imprévus pour ensuite proposer de petites améliorations pouvant atténuer les déficiences que LoRaWAN peut connaître. Les déficiences identifiées par les auteurs sont multiples, telles que : l'impossibilité de transmettre et de recevoir simultanément sur le même canal dans une gateway, l'utilisation du SF12 (bas débit, ToA élevé) sur la deuxième fenêtre de réception (RX2) menant à une atténuation rapide de la restriction de Duty Cycle des gateways (10%), la transmission DL sur RX1 utilisant le même canal que la transmission UL peut être en concurrence avec d'autres transmissions UL générant des interférences.

L'étude faite par les auteurs a permis d'étudier l'impact de certains paramètres afin de les contrôler et atténuer leurs effets. Parmi les points étudiés :

— Activation (DC on) et désactivation (DC off) du Duty Cycle de la gateway (limité à 10% dans la bande ISM 867) dans le but d'analyser son impact sur les performances du réseau,

— Priorisation de l'émission (TX) (dans ce cas, tous les RX seront interrompus lors de la transmission TX) ou de la réception (RX) (Dans ce cas, tous les TX seront reportés jusqu'à réception de RX) dans le but de minimiser les interférences et d'augmenter le taux de réception,

— Échange des sous-bandes (Sub-bands swapping) : les auteurs ont activé un mode qui permet d'ouvrir RX1 sur le canal DL dédié (869.525 MHz) et RX2 sur le canal utilisé pour la transmission UL (le contraire de ce qui est implémenté dans la norme LoRaWAN),

— Augmentation du débit de l'ACK (ADR Data Rate ACK) : contrairement à la norme LoRaWAN qui propose que les transmissions ACK sur RX1 utilisent le même SF que la transmission UL et que les transmissions ACK sur RX2 utilisent SF12, les auteurs modifient le module LoRaWAN de sorte que les ACKs transmis sur RX1 et RX2 utilisent un plus haut débit (SF i $<$ SF 12) dans le but de minimiser la probabilité d'atteindre le Duty Cycle de gateway rapidement,

— variation du nombre de re-transmissions : les auteurs proposent de varier le nombre de re-transmissions maximal en prenant les valeurs dans l'ensemble $\{1, 2, 4, 6, 8\}$ dans le but de voir l'impact de ce paramètre sur la performance du réseau,

— Utilisation de deux gateways séparées : les auteurs proposent d'implémenter une gateway spécifique pour la transmission des DL et une autre spécifique pour la réception des UL afin de minimiser les interférences (Full Duplex GW),

— variation du nombre de canaux : les auteurs proposent de varier le nombre de canaux disponibles pour analyser l'impact de cette variation sur le réseau.

Les auteurs dans un premier temps ont présenté une analyse systématique de l'impact de ces paramètres réglables sur deux métriques, à savoir l'UL-PDR (probabilité qu'une transmission UL soit reçue correctement) et le CPSR (probabilité qu'une transmission UL ainsi que son acquittement soient reçus correctement). Tout d'abord, les auteurs ont observé qu'avec une configuration de paramètres standard (par défaut), dans un scénario mixte (Trafic confirmé + non confirmé) la présence des trafics confirmés peut dégrader considérablement les performances des trafics non confirmés, en raison des interférences supplémentaires générées par les transmissions DL (ACK). D'autre part, en considérant seulement les trafics confirmés, alors, le facteur le plus critique semble être la contrainte DC de la gateway. En effet, le canal DL devient rapidement le goulot d'étranglement du système en présence de flux bi-directionnels.

Dans un second temps, les auteurs ont observé qu'en modifiant légèrement la procédure ACK (à savoir, en introduisant les mécanismes Sub-bands swapping et ADR Data Rate ACK) et en donnant la priorité à la réception RX par rapport à la transmission TX au niveau de la gateway (ou, même mieux, en permettant la priorisation sélective de certaines transmissions DL), il est possible d'améliorer sensiblement les performances en termes de taux de réception de paquet, de capacité du système, d'efficacité énergétique et d'équité, en particulier en présence des sources de trafic mixtes. Inversement, d'autres paramètres du système, comme le nombre maximum de tentatives de transmission et le nombre de voies parallèles reçues, semblent être déjà bien configurés et dimensionnés dans la norme LoRaWAN.

La proposition des auteurs est intéressante, mais pourrait être nettement améliorée en la combinant avec une stratégie d'adaptation de débit des transmissions UL (amélioration du ADR Basique). Les SFs utilisés dans les deux fenêtres RX1 et RX2 dépendent du SF choisi par le serveur pour la transmission UL et que l'élément impactant le plus sur la réduction du taux de paquets reçus dépend de la configuration utilisée par les noeuds pour la transmission UL avant d'arriver à la sélection de paramètres DL. Donc une stratégie d'allocation de débit optimale pour les transmissions UL contribuera à l'amélioration de la contribution des auteurs et à l'augmentation du taux de paquets reçus.

2.3 Choix des paramètres de configuration LoRa basé sur le sondage

Dans [34], les auteurs se sont intéressés à la combinaison des paramètres de transmission LoRa que nous avons présentés dans le chapitre 1. En fait, le nœud LoRa peut être configuré pour utiliser différents paramètres SF, BW, CR et TP, ce qui donne un total de 6720 combinaisons possibles. Les auteurs ont confirmé qu'il peut exister un certain nombre de configurations offrant une meilleure qualité de liaison, au prix d'une consommation d'énergie plus élevée (une augmentation de la consommation d'énergie d'un facteur supérieur à 100). Ils ont donc étudié comment déterminer un réglage approprié qui minimise la consommation d'énergie tout en répondant à l'exigence de performance de communication en termes de PRR. Ils ont proposé un protocole qui explore périodiquement différentes combinaisons de configuration au moyen d'un sondage et choisit dynamiquement ceux qui minimisent la consommation énergétique. Pour ce faire, des paquets sont envoyés à partir d'une portée fixe en variant les réglages des paramètres afin de calculer le PRR pour chaque combinaison. Sur la base de ces PRR, ils déterminent les paramètres satisfaisant à l'exigence de performance pour un seuil PRR donné après un certain nombre de sondes (itérations). Considérant que la

meilleure configuration est celle dont la valeur RSSI correspond aux intervalles présentés dans Table II.1 qu'on nommera $Conf_{best}$, les auteurs ont évalué et comparé les résultats par rapport à cette meilleure configuration ($Conf_{best}$). Les résultats montrent qu'après 16 transmissions réussies et 285 sondes (transmissions totales), l'algorithme trouve une configuration qui ne consomme que 44% d'énergie en plus que $Conf_{best}$. Contrairement à la configuration allouée par l'ADR de base qui consomme beaucoup plus que 44% par rapport à $Conf_{best}$. Le principal inconvénient de cette solution est que le test des différentes configurations inadéquates (pendant 16 transmissions) entraîne soit une surconsommation d'énergie, soit des pertes de paquets si ces derniers ne pouvaient pas atteindre la gateway. De plus, la complexité est élevée vu le nombre d'itérations (285 transmissions totales pour 16 réussites). Son utilisation pratique dans des conditions de réseau dynamiques (Par exemple, interférences, mobilité des nœuds, etc.) peut donc ne pas être efficace en raison de la nécessité de recalculer le nouveau seuil de PRR pour chaque nouvelle condition de réseau.

2.4 Contrôle adaptatif du débit orienté noeud basé sur un contrôle de congestion du réseau

Une autre problématique liée à l'ADR concerne la perte des accusés de réception (ACK). En effet, quand un équipement ne reçoit pas d'ACK, il juge que c'est un problème de portée et augmente son SF. Cependant, le fait que l'ACK n'est pas reçu peut également être dû à la congestion du réseau. Dans ce cas, la diminution du débit par l'augmentation du SF peut entraîner une augmentation du ToA (provoquant des collisions plus importantes) et une surconsommation d'énergie. Ce problème a été abordé dans [50] en identifiant d'abord la cause de la perte de l'ACK et en ajustant ensuite le temps de transmission back-off du nœud en cas de congestion, au lieu d'ajuster le SF. Tout d'abord, afin de détecter les situations de congestion, les auteurs ont proposé un classificateur de congestion qui fait un apprentissage sur l'état de la connexion sans fil en utilisant une méthode de régression logistique. Le classificateur de congestion prend en entrée le débit, le RSSI et le nombre de gateways. Il fonctionne en deux parties : nœud et serveur. Le serveur est responsable de la collecte des exemples d'apprentissage et de l'application de l'algorithme d'apprentissage supervisé (régression logistique) pour trouver les meilleurs poids. Ces meilleurs poids seront partagés et mis à jour périodiquement. Les nœuds, sur la base des poids diffusés par le serveur, classifient l'état d'encombrement Y ($Y \in \{0, 1\}$). Ensuite, une fois la congestion classée, le nœud compte le nombre de messages envoyés (ADR-MSG-CNT) ainsi que le nombre d'accusés de réception reçus (RCV-ACK-CNT). S'ils sont égaux, le DR sera incrémenté. Sinon, le nœud attend un délai (ADR-ACK-Delay) puis vérifie l'état de la congestion pour décider soit d'incrémenter le délai d'attente (dans le cas où $Y = 1$), soit de diminuer le DR (dans le cas où $Y = 0$). L'avantage de cette approche est qu'elle tient compte du niveau de congestion du réseau, contrairement à l'ancien système ADR, ce qui minimise les changements de DR inefficaces. Par conséquent, l'inconvénient est que le processus exige un message ACK pour chaque transmission. Comme le trafic DL a un effet négatif sur le débit de l'UL, le PRR diminue en conséquence. En outre, si un ACK n'est pas reçu, le nœud doit retransmettre le paquet jusqu'à 64 fois avant de décider de modifier la configuration. Ainsi, à cause du long temps de retransmission, et du délai de collecte des données nécessaires à l'apprentissage, ce mécanisme ne semble pas adapté pour traiter des réseaux dynamiques avec des nœuds

mobiles.

2.5 Allocation adaptative équitable du débit de données et contrôle de la puissance dans LoRaWAN

L'analyse effectuée par les auteurs dans [51] a montré que le choix local du SF et du TP par les nœuds entraîne un réseau inéquitable avec un taux de perte de paquet élevé pour les nœuds éloignés de la station de base. En fait, les paquets utilisant le même SF s'interfèrent résultant en un SNR erroné. Dans ce dernier cas, les nœuds éloignés de la station de base, qui subissent beaucoup plus de collisions, augmentent leur SF, ce qui aggrave la situation puisque le temps d'occupation du canal augmente et la probabilité d'interférence s'élève. En conséquence, les auteurs ont proposé un algorithme de contrôle de la puissance et du facteur d'étalement équitable appelé « FADR » pour atténuer cet effet. L'objectif est d'obtenir un débit de données équitable et de réduire les collisions entre les nœuds, ce qui permet d'augmenter le PRR. La première contribution est le calcul de la distribution optimale du SF entre les nœuds pour minimiser la probabilité de collision. La seconde est un schéma qui distribue le SF et les paramètres TP discrets aux nœuds. Les auteurs supposent qu'au départ, le TP est le même pour tous les nœuds et que les RSSI des premières transmissions seront utilisés comme paramètres d'entrée dans l'exécution du FADR. Après $n = 20$ transmissions, les nœuds sont ordonnés en fonction de leurs RSSI et divisés en groupes de 50 nœuds. Dans chaque groupe, le SF12 est attribué à un seul nœud (ayant le plus petit RSSI), les autres SF sont répartis entre les 49 nœuds restants par groupe en fonction de leurs valeurs de RSSI. L'équité de la distribution des ressources (SF) est assurée par le schéma d'allocation de puissance proposé, qui vise à optimiser le RSSI des nœuds, en allouant une faible TP aux nœuds ayant un signal fort et une TP élevée aux nœuds ayant un signal faible. Ils permettent d'obtenir un PRR uniforme pour tous les nœuds d'extrémité malgré la distance de la passerelle et de maintenir la durée de vie du nœud d'extrémité en appliquant des niveaux de TP faibles. Ce qui permet d'égaliser le niveau du RSSI des noeuds, c'est-à-dire l'augmentation du niveau RSSI en augmentant la TP [59], et ainsi d'atténuer l'effet de capture et l'orthogonalité imparfaite des SF. Les résultats expérimentaux montrent que plus le SF7 est utilisé, plus le PRR et l'efficacité énergétique sont élevés (environ 22% de consommation d'énergie en moins), grâce au fait que l'allocation de puissance a égalisé la puissance des nœuds au plus bas tout en la contrôlant pour minimiser les collisions dans le cas de l'utilisation de grands SF. De plus, les auteurs ont montré qu'en augmentant le nombre de nœuds dans le réseau, l'indice d'équité entre les nœuds diminue. Cela favoriserait la communication des nœuds d'extrémité les plus proches de la passerelle par rapport à ceux qui en sont plus éloignés. Cette approche n'est donc possible que dans des réseaux extrêmement petits dont les nœuds sont positionnés près de la passerelle. En outre, bien que le FADR améliore le PRR de base, son adaptation aux changements d'état du réseau (par exemple, en raison de la mobilité des nœuds) est faible en raison de la nécessité d'attendre n transmissions avant la réadaptation (dans le cas où n est grand).

2.6 Adaptation du débit en fonction de la qualité moyenne du signal

Dans [4], les auteurs ont évalué les performances du ADR Basique proposé par Semtech et ont confirmé que le mécanisme est efficace pour augmenter le PRR et diminuer la consommation d'énergie dans des conditions stables. Cependant, ils ont indiqué que l'ADR est gravement affecté par un canal sans fil extrêmement variable, comme la présence d'obstacles mobiles qui entraînent la variation des valeurs du SNR et du RSSI d'où proviennent les erreurs d'allocation de la configuration. Ils ont également montré que la valeur maximale du RSSI des 20 derniers paquets est une approche inadéquate pour estimer la qualité du lien pour un canal variable, suite à des conditions météorologiques variables ou des obstacles temporaires devant les nœuds de communication. Les auteurs ont donc proposé une amélioration du ADR Basique appelée « ADR+ », mise en œuvre du côté serveur afin de permettre aux noeuds de mettre à jour dynamiquement leurs paramètres (SF, TP). Au lieu de considérer la valeur maximale de RSSI des 20 derniers paquets comme dans le cas de ADR basique, les auteurs ont proposé une stratégie d'allocation du SF basée sur la valeur moyenne des RSSI des 20 derniers paquets. Le système proposé a montré que la configuration appropriée des SF et des TP pourrait augmenter la capacité du réseau et réduire l'utilisation de l'énergie, sur la base des connaissances générales du réseau. ADR+ a permis d'améliorer de 30% le PRR par rapport à ADR basique dans le cas de conditions du canal variables modérées. Il a montré son efficacité dans le cas où le nombre de paquets ayant des valeurs RSSI faibles est supérieur à celui ayant des valeurs plus élevées. Toutefois, dans le cas contraire, le PRR n'est pas amélioré. De plus, le cas de la mobilité n'est pas explicitement pris en compte. Par ailleurs, attendre 20 paquets pour ajuster le schéma peut être trop long pour s'adapter rapidement à des changements fréquents de l'environnement.

2.7 Adaptation de débit en utilisant tous les SF

Les auteurs de [5] cherchent à trouver un moyen d'allouer les ressources de manière à prendre en charge un grand nombre de nœuds dans la même zone en minimisant le taux de collision. Ils ont confirmé que l'ADR dans LoRaWAN ne tient pas compte de la répartition du SF sur le nombre de nœuds et qu'il existe un risque d'attribuer le même SF à tous les nœuds connectés si tous les RSSI maximaux reçus correspondent à une valeur unique du SF. Les auteurs ont constaté que bien qu'un SF élevé corresponde à une longue portée de couverture radio qui permet à un grand nombre de paquets d'arriver correctement, le canal sera occupé pendant longtemps, ce qui se traduira par un taux de collision croissant, donc un PRR faible. Ils ont proposé d'utiliser toutes les valeurs de SF disponibles pour tous les nœuds couverts par la gateway afin de diminuer le taux de collision et d'améliorer le débit. Pour ce faire, les auteurs ont proposé deux approches : « EXPLoRA-SF » et « EXPLoRA-AT ».

2.7.1 EXPLoRA-SF

L'approche « EXPLoRA-SF » vise à répartir les nœuds de manière égale sur le nombre de valeurs SF disponibles afin de réduire le risque de collision. Six groupes correspondant aux six valeurs SF sont créés. Chaque groupe se verra attribuer des noeuds en fonction de

leurs valeurs RSSI. Pour ce faire, lors de la phase d'enregistrement, le serveur collecte les valeurs RSSI de toutes les demandes (Join Request). Ces valeurs formeront une matrice. Les noeuds sont ordonnés selon leurs valeurs RSSI dans l'ordre décroissant puis divisés en $n = 6$ groupes (de même nombre de noeuds). Le premier groupe de nœuds situé près de la gateway, sur la base des valeurs RSSI les plus élevées, se verra attribuer le SF le plus bas (SF7), et ainsi de suite jusqu'au dernier groupe de nœuds. Le dernier groupe se verra attribuer le SF le plus élevé (SF12) car les nœuds de ce groupe se trouvent loin de la gateway. Si nous avons $N = 6$ nœuds connectés, chaque groupe ne contiendra qu'un seul nœud où chaque nœud aura un SF différent. Lorsque le nœud se verra attribuer un SF, sa valeur RSSI sera retirée de la matrice, et ainsi de suite jusqu'à ce que tous les nœuds aient leur configuration SF et que la matrice contienne 0 élément. Bien que la distribution proposée permette au réseau d'utiliser tous les SF disponibles pour minimiser les collisions, elle ne tient pas compte des limites des intervalles RSSI prévus pour chaque cas. Dans le cas où la valeur RSSI (-120 dB) par exemple est la plus grande valeur parmi toutes celles reçues, le serveur attribue le nœud SF7 qui n'est pas adéquat puisque la limite RSSI du SF7 est de (-117 dB) (Table II.1). Cela entraînera des pertes de paquets. Un autre cas peut se produire, où le plus petit RSSI est égal à (-125 dB), et le serveur attribue un SF12 dont la limite RSSI est égale à (-134 dB). Dans ce cas, on constate une sur-utilisation de configuration et donc une surconsommation d'énergie.

2.7.2 EXPLoRA-AT

Le deuxième schéma proposé, « EXPLoRA-AT », répartit les différents SFs en égalisant le ToA entre les différents nœuds. Cette stratégie optimisera l'utilisation des différentes valeurs de SF au fil du temps, en visant une répartition égale de l'occupation des canaux en assurant l'égalité du ToA. Si le nœud A doit transmettre 10 paquets de 40 octets et qu'un autre nœud B envoie 4 paquets de 5 octets, le nœud A aura un petit SF qui lui permettra d'envoyer rapidement ses données avec une DR plus élevé et le nœud B aura une valeur de SF plus importante et transmettra ses données en moins de temps. Bien que cette approche permette une certaine équité, les distances entre les nœuds et les gateways ne sont pas prises en compte. Si le nœud A est éloigné de la gateway, l'utilisation d'un SF faible entraîne une perte de paquets. De plus, si le nœud B est proche de la gateway, le SF élevé alloué augmente la consommation d'énergie en augmentant le ToA.

Lorsque le réseau est fortement chargé, l'allocation du SF doit tenir compte des collisions. Pour cela, EXPLoRA-AT vise à égaliser le temps d'occupation du canal de tous les noeuds en égalisant leurs ToAs grâce à l'utilisation de différents SFs.

2.8 Approche méta-heuristique pour l'optimisation des paramètres de transmission

Un modèle mathématique dont l'analyse numérique aboutit à une formule qui maximise le débit tout en respectant la réglementation sur le cycle d'utilisation de la transmission (TDC) a été proposé dans [52]. Ce modèle permet de pallier les restrictions strictes imposées sur le cycle d'utilisation des bandes ISM dans certaines régions et d'améliorer l'efficacité de la transmission et de la puissance dans les nœuds finaux. L'approche présente deux méta-heuristiques pour résoudre un problème d'optimisation en calculant la politique de

transmission. Une politique de transmission optimale signifie une sélection optimale de BW, SF, CR et TP. En considérant conjointement l'utilisation de la bande et l'efficacité de l'utilisation de la puissance, la performance maximale des nœuds en termes de débit est calculée en utilisant un modèle de Markov et est optimisée par rapport à un ensemble de paramètres de transmission possibles, obtenant ainsi ce que l'on appelle la politique de transmission optimale. Cette méthode augmente la performance de plus de 33% par rapport au schéma ADR Basique et est indiquée pour pouvoir fonctionner dans des dispositifs IoT à contrainte énergétique. L'avantage de cette approche est que le modèle est basé sur une théorie de dérivation de politique optimale, qui donne à l'approche la capacité de prédire statistiquement la performance des nœuds dans le réseau LoRaWAN. L'inconvénient est qu'il s'agit d'un problème d'optimisation combinatoire complexe qui ne peut être résolu directement. De plus, la mobilité n'est pas considérée dans cette étude.

2.9 Critique

L'idée principale de l'ADR Basique et de ses améliorations citées ci-dessus est de permettre au serveur LoRa de trouver la configuration optimale des noeuds selon les conditions du réseau, ayant comme objectif principal l'augmentation du débit. La plupart de ces propositions atteignent cet objectif en observant les RSSIs / SNRs reçus par le serveur et en recalculant à travers ces valeurs (en utilisant différents algorithmes selon la variante ADR) une nouvelle configuration. Chaque variante diffère des autres dans les critères de décision pour changer la configuration (nombre de paquets reçus, valeurs RSSI, etc.). En général, les noeuds utilisent au début le plus grand SF pour s'assurer d'atteindre les gateways, ensuite le serveur ré-attribue une nouvelle configuration en diminuant le SF pour maximiser le DR et minimiser la consommation énergétique. La majorité des variantes proposées s'intéressent à des noeuds statiques situés dans un environnement stable. Quelques travaux ont traité le cas des environnements variables en tenant compte des changements dans les conditions météorologiques ou le passage de certains obstacles. Cependant, très peu de travaux ont traité le cas de la mobilité des noeuds en se limitant à montrer la non-adaptation de ADR Basique dans ces conditions.

D'autre part, on peut remarquer, pour toutes les variantes ADR, que le SF peut seulement être diminué du côté serveur (ADR-Server) et que son augmentation n'est assurée qu'après un long processus de retransmissions du côté du noeud (ADR-Node). Ce qui limite leur adaptation (réadaptation) rapide aux changements de conditions. En outre, les retransmissions excessives peuvent rapidement conduire le noeud à dépasser le Duty Cycle de 1% entraînant également à une perte de paquets, puisque les paquets qui n'ont pas été transmis suite à ce dépassement seront tout simplement considérés comme perdus.

Afin de tenir compte de ces contraintes (mobilité, duty cycle, etc.) nous proposons une amélioration de ADR appelée "E-ADR".

Cette amélioration vise à s'adapter à tout type d'environnement entre autres les obstacles et la mobilité tout en assurant une bonne fiabilité et une meilleure efficacité énergétique. La section suivante détaille le mécanisme E-ADR.

3 – E-ADR

Le choix d'une configuration optimale des paramètres de transmission a un impact direct sur la portée de la transmission, le ToA et la consommation d'énergie. L'objectif d'E-ADR est de garantir une transmission fiable (PRR élevé) face aux changements dynamiques de l'état du réseau dus à la mobilité des nœuds, tout en minimisant la consommation d'énergie (par la réduction du ToA). E-ADR est exécuté du côté du serveur, où l'incrémentation et la décrémentation du débit est proposé pour répondre aux besoins du noeud en termes de QoS.

Contrairement aux autres variantes ADR qui réagissent aux changements des valeurs RSSI pour diminuer le SF dans la partie serveur (ADR server) et aux paquets perdus pour incrémenter le SF dans la partie noeud (ADR node), l'idée principale d'E-ADR est de régler d'une manière précise le SF en autorisant le serveur à l'incrémenter et le décrémenter selon les futures positions du noeud prédites à partir des positions courantes des noeuds apprises à travers les valeurs RSSI des paquets reçus. La prédiction des prochaines positions des noeuds est faite, dans un premier temps, selon la technique de régression linéaire pour les applications dont la trajectoire est connue. Dans la deuxième partie du chapitre, une extension d'E-ADR est proposée pour supporter les applications à trajectoire inconnue. Cette extension, appelée "VHMM-based E-ADR", se base sur le modèle de Markov caché d'ordre variable. E-ADR, avec ses deux versions, est considéré comme un mécanisme proactif permettant une rapide adaptation à la différence des autres variantes ADR de la littérature qui sont réactives.

3.1 E-ADR pour les modèles de mobilité connus

E-ADR est conçu pour des applications avec des modèles de mobilité connus où les nœuds mobiles (robots et drones pour le domaine de l'agriculture, par exemple [48]) sont programmés pour exécuter des tâches suivant une trajectoire prédéfinie avec des transmissions de paquets périodiques et une vitesse de déplacement fixe. Chaque nœud rejoint le serveur en envoyant une requête comprenant l'identifiant de l'application (ID). Cet ID permet au serveur de déterminer le modèle de mobilité qui sera suivi par le nœud (modèle de mobilité prédéfini). Le nœud commence par transmettre n paquets avec le $mode1$ (Table II.1), ensuite en utilisant l'algorithme de trilatération [60, 61], la position du noeud pour chaque paquet transmis est calculée en utilisant au moins 3 gateways. À partir des positions calculées, le serveur prédit les n positions suivantes du nœud mobile à l'aide d'un algorithme de prédiction par régression linéaire basé sur la zone et la trajectoire prédéfinies, et utilise la position la plus lointaine du gateway pour calculer le RSSI correspondant. Ainsi, il détermine le nouveau mode correspondant à l'intervalle RSSI auquel appartient la valeur calculée (Table II.1), puis annonce la nouvelle configuration au noeud concerné.

3.1.1 Techniques de localisation

La détermination précise de la position des nœuds émetteurs est une question vitale, car les données recueillies sont étroitement liées aux informations de localisation. Différentes techniques de localisation peuvent être utilisées : GPS [62], les systèmes de localisation des réseaux cellulaires [63], localisation par infrarouge [64], localisation par ondes ultra-sonores [65], etc. Une caractéristique commune à ces stratégies de positionnement est que des modules

supplémentaires sont intégrés dans les nœuds de capteurs, ce qui entraîne une augmentation de la consommation d'énergie, mais aussi du coût de déploiement. Pour résoudre ce problème, une solution est d'acquérir la localisation sans dispositifs supplémentaires en utilisant uniquement les informations du signal reçu qui peuvent être disponibles dans l'émetteur-récepteur radio utilisé pour la communication. Ces informations peuvent être l'heure d'arrivée, le décalage horaire d'arrivée (TDOA), l'angle d'arrivée (AOA) et l'indicateur de l'intensité du signal reçu (RSSI).

L'auteur dans [66] a utilisé l'approche de trilatération qui consiste à définir la position du noeud à partir des informations collectées à partir de trois gateways. Chaque gateway définit la distance qui la sépare du noeud en se basant sur le RSSI du paquet reçu. La distance entre un noeud et une gateway représente le radium du cercle ayant pour centre la position de la gateway. L'intersection des trois cercles des gateways représente la position ajustée du noeud. La Commission européenne [67] a utilisé la technique de l'angle d'arrivée (AoA) où la position d'un nœud est estimée par 2 stations de base équipées d'un réseau d'antennes. La technique de la différence de temps d'arrivée (TDOA) [68] utilise trois gateways ou plus avec des références temporelles précises.

Par rapport aux autres méthodes, l'algorithme de trilatération peut calculer la position directement et rapidement à condition que trois gateways soient placées de manière non-linéaire. L'algorithme de trilatération est une méthode de positionnement de base, largement utilisée dans de nombreux systèmes de localisation [69]. La relation entre la position du nœud inconnu et les trois positions des gateways peut être exprimée comme suit [70] :

$$\begin{cases} (x-x_1)^2+(y-y_1)^2=d_1^2 & \text{(II.1a)} \\ (x-x_2)^2+(y-y_2)^2=d_2^2 & \text{(II.1b)} \\ (x-x_3)^2+(y-y_3)^2=d_3^2 & \text{(II.1c)} \end{cases}$$

où (x,y) sont les coordonnées du noeud inconnu, ($x1$,$y1$), ($x2$,$y2$), ($x3$,$y3$) sont les coordonnées des 3 gateways et (d_1, d_2 et d_3) sont les distances entre le noeud et chacune des gateways, obtenues à partir des différents RSSI. L'équation II.1 peut être écrite sous forme matricielle :

$$\mathbf{Qx} = \mathbf{b} \tag{II.2}$$

où $\mathbf{Q}$ est une matrice de dimension $(r \times r)$, $\mathbf{x}$ est le vecteur des coordonnées du noeud inconnu, $\mathbf{b}$ est le vecteur de dimension r, et r est la dimension des coordonnées du noeud inconnu. Pour des coordonnées $2D$, $\mathbf{Q}$ est de dimension (2×2) et $\mathbf{b}$ de dimension 2 :

$$\mathbf{Q} = \begin{bmatrix} 2(x_1-x2) & 2(y_1-y2) \\ 2(x_1-x3) & 2(y_1-y3) \end{bmatrix} \tag{II.3}$$

$$\mathbf{b} = \begin{bmatrix} x_1^2-x_2^2+y_1^2-y_2^2+d_1^2-d_2^2 \\ x_1^2-x_3^2+y_1^2-y_3^2+d_1^2-d_3^2 \end{bmatrix} \tag{II.4}$$

Le fait que $\mathbf{Q}$ soit ou non-inversible dépend de sa valeur déterminante. Un bon choix des positions des gateways peut garantir que la matrice soit inversible, d'où la condition des gateways placées de manière non-linéaire. Par conséquent, la position estimée est donnée par Eq. II.5 où ($\tilde{x}$,$\tilde{y}$) sont les coordonnées de la position estimée du noeud inconnu :

$$\mathbf{x} = \mathbf{Q}^{-1}\mathbf{b} = \begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix}$$

(II.5)

Dans notre travail, nous avons choisi la technique de trilatération pour sa simplicité. Nous supposons que les positions des gateway sont connues. En utilisant la valeur RSSI du paquet reçu, nous déduisons la distance séparant le nœud de chaque gateway à l'aide de l'Eq. II.6 [71].

$$Distance = 10^{((MeasuredPower - RSSI)/(20))} \tag{II.6}$$

Il est communément admis que l'estimation de la distance basée sur le RSSI souffre d'un problème d'imprécision. Cependant, comme il suffit d'ajuster les paramètres de transmission LoRa en fonction du RSSI (c'est-à-dire la distance électromagnétique), mais pas la distance géographique, ce problème d'imprécision a peu d'impact sur l'E-ADR.

3.1.2 Prédiction de la prochaine position -Régression linéaire-

Pour choisir la bonne configuration, il est nécessaire de prévoir les prochaines positions du nœud mobile. Pour ce faire, nous avons choisi la méthode de régression linéaire [72] en raison de sa faible complexité et de son utilisation largement éprouvée dans les algorithmes d'apprentissage automatique supervisé pour la modélisation prédictive. La position future du nœud est calculée sur la base des n positions précédentes $P_i(Xi,Yi)$ et selon le modèle suivi en tenant compte de la zone d'intérêt (AoI) que nous surveillons. Nous supposons que le changement de configuration doit être effectué à chaque n paquets ($1 < n < 20$). Un choix de n très petit (égal à 1) génère un sur-coût important lors du changement de configuration, tandis que n élevé (par exemple 20) provoque la perte des paquets envoyés avec des paramètres inappropriés, d'où le choix de $n = 3$.

Tout d'abord, le serveur calcule la variation moyenne sur la base des n positions précédentes en utilisant Eq. II.7.

$$Avg_{variation}(X_{Avg}, Y_{Avg}) = \frac{\sum_{i=1}^{n}(P_{i+1} - P_i)}{n} \tag{II.7}$$

A partir de cette variation moyenne $Avg_{variation}(X_{Avg}, Y_{Avg})$, le serveur estime les positions suivantes en utilisant Eq. II.8 en vérifiant à chaque fois si ces positions estimées ne sortent pas de l'AoI (X_{min} , X_{max}, Y_{min} , Y_{max}), sinon il tourne à gauche ou à droite / vers le bas ou vers le haut selon l'emplacement estimé du nœud. La position (X_n,Y_n) est la dernière position estimée calculée par la trilatération.

$$P_{n+1}(X_n + X_{Avg}, Y_n + Y_{Avg}) \tag{II.8}$$

3.1.3 Modèle d'allocation de configuration

Une fois la nouvelle position estimée, le serveur estime les distances entre le nœud et les gateways pour les $n = 3$ prochains paquets et les convertit en valeurs RSSI estimées en utilisant Eq. II.9 [71]. Ensuite, le serveur choisit la valeur RSSI qui correspond à la gateway la plus distante afin de garantir que le noeud soit dans la couverture des gateways. Enfin, il vérifie si la valeur RSSI estimée appartient à l'intervalle RSSI correspondant au mode de configuration actuel (les limites RSSI sont présentées dans Table II.1) ou non, pour décider si la configuration doit être modifiée ou non.

$$RSSI = MeasuredPower - 20log(Distance) \tag{II.9}$$

Dans le cas où la valeur RSSI estimée n'appartient pas à l'intervalle RSSI associé au mode actuel, un taux de transition $R_{a\to b}$ du mode a vers le mode b sera calculé pour définir le mode approprié à configurer selon l'Eq. II.10.

$$R_{a\to b} = \frac{1/2 \times |B_{sup}(b) - B_{inf}(a)|}{|B_{sup}(b) - RSSI_{estim}(n+1)| + |B_{inf}(b) - RSSI_{estim}(n+1)|} \tag{II.10}$$

Ce taux définit le degré d'appartenance du RSSI estimé à l'intervalle $[B_{inf}, B_{sup}]$ de chaque mode de configuration. B_{inf} définit la limite inférieure de l'ancien mode (a) et B_{sup} est la limite supérieure du mode prédit (b). Enfin, le serveur choisira le mode ayant le taux le plus élevé dans le vecteur présenté dans l'Eq. II.11. La nouvelle configuration (mode) sera communiquée au noeud par le message de commande $LinkADRReq$ (CID=$0x03$) [21].

$$R_{a\to b} = \max(R_{a\to 1}, R_{a\to 2}, \cdots, R_{a\to 9}, R_{a\to 10}) \tag{II.11}$$

3.2 Extension d'E-ADR pour les modèles de mobilité inconnus « VHMM-based E-ADR »

Dans un environnement intelligent, il existe plusieurs applications mobiles telles qu'un robot de surveillance, des colliers et des puces connectées pour le suivi des animaux, etc. Contrairement aux robots et aux drones dédiés à la surveillance des parcelles et des plantations qui suivent une trajectoire connue, les capteurs destinés au suivi des animaux suivent une trajectoire inconnue. Dans ce cas, E-ADR n'est plus adapté puisqu'il se base dans son calcul de la position sur un modèle de mobilité prédéfini pour chaque noeud. Dans cette partie, nous proposons une extension d'E-ADR afin qu'il s'adapte au cas des noeuds mobiles à trajectoire inconnue. Pour cela, nous proposons d'intégrer un schéma de prédiction de la mobilité basé sur le modèle de Markov caché d'ordre variable (Variable HMM-(2,1)). Grâce au résultat de prédiction de la prochaine localisation probable du noeud, le serveur LoRa peut lui proposer une meilleure configuration.

Ces dernières années, la prédiction de la trajectoire des noeuds mobiles est devenue un sujet de recherche important. De nombreuses méthodes de prédiction sont utilisées pour déterminer la position d'un mobile.

[73] a proposé la probabilité de transition de Markov, qui est basée sur une organisation cellulaire d'un espace récepteur, et l'a utilisé pour extraire des statistiques de mobilité de bases de données spatio-temporelles indexées. [74] a utilisé les modèles de Markov cachés (HMM) pour extraire des modèles de trajectoires en utilisant l'efficacité des méthodes de partitionnement de l'espace. Un autre travail de [75] a utilisé des chaînes de Markov cachées pour prédire les trajectoires d'objets en mouvement, dans lequel les trajectoires historiques sont transmises dans des graphiques dirigés pour construire une chaîne de Markov, qui est utilisée pour calculer la matrice de transition d'ordre prédéfini γ et prédire la trajectoire de déplacement. Toutefois, l'utilisation du processus pour un ordre γ faible ne garantit pas la précision de la prédiction et inversement, le choix d'un ordre élevé γ rendra le processus très complexe. De même, dans [76], les auteurs ont utilisé la corrélation spatiale de la disposition des stations de base pour améliorer la probabilité partielle du processus de solution dans les HMM, qui travaille à restaurer la séquence de trajectoire sans prendre en compte les états observables manquants. Cependant, si l'ordre des états γ est faible, les résultats de la prédiction seront grandement affectés.

[77] a récemment étendu le modèle de la chaîne de Markov à la mobilité en prenant en compte l'effet de la visite d'un lieu avant un nombre γ d'états, ce qui ressemble davantage à une chaîne de Markov d'ordre élevé. Cependant, la plupart des modèles de prédiction basés sur le modèle Markov ne prennent pas en compte la chaîne d'états cachés d'ordre élevé à cause de leur complexité et les problèmes de rétention d'états continuent à affecter grandement la précision des prédictions. Dans [78], les auteurs constatent qu'un modèle de Markov caché d'ordre $\gamma = 2$ (HMM-2) est plus performant que d'autres prédicteurs et qu'il est simple à mettre en œuvre. En outre, les auteurs de [79] et [80] proposent un schéma de prédiction optimisé basé sur le HMM. Ils constatent que la prédiction de la mobilité utilisant HMM donne de meilleurs résultats. Cependant, leurs travaux sont basés sur l'algorithme de Viterbi [81], qui est coûteux, tant en termes de mémoire que de temps de calcul. Pour une séquence de longueur γ, la programmation dynamique pour trouver le meilleur chemin dans un modèle avec s états et e edges prend de la mémoire proportionnelle à $s \times \gamma$ et du temps proportionnel à $e \times \gamma$.

Nous notons que le modèle de Markov caché d'ordre γ a été tant utilisé pour la prédiction des déplacements résultant en une très grande efficacité en termes de prédiction, de temps de calcul et de mémoire lorsque l'ordre γ est bien choisi. D'après l'analyse de [82], il est évident que contrairement aux autres prédicteurs, HMM peut améliorer les performances de la prédiction de la mobilité, car il réduit les pertes d'informations. Le modèle de prédiction HMM-γ (ordre γ) est le modèle le plus utile et le plus facile à implémenter, par contre l'ordre HMM doit être bien choisi en fonction de la phase d'apprentissage dont on dispose, afin d'avoir moins de complexité et d'efforts d'apprentissage et une meilleure précision.

Dans notre travail, nous nous intéressons à la prédiction du prochain emplacement du noeud mobile LoRa en utilisant un modèle de Markov caché d'ordre variable (2,1) (VHMM-(2,1)).

3.2.1 Variable HMM-(2,1)

Un HMM d'ordre γ est constitué d'un ensemble fini d'états (variables cachées), d'une séquence de symboles d'émission ou de sortie (variables observables), d'un ensemble fini de probabilités de transition d'état et d'un ensemble de probabilités d'émission. Le modèle

Variable HMM repose sur la combinaison de deux HMM d'ordres différents, par exemple Variable HMM-(2,1) combine HMM-1 et HMM-2. À la différence du HMM-1, dans lequel l'état actuel POI(j) détermine le suivant POI(k) , le HMM-2 dépend de son état actuel POI(j) et de l'état précédent POI(i) pour déterminer la prochaine position k (POI(k)).

Le modèle Variable HMM-(2,1) (VHMM-(2,1)) repose sur l'entraînement et la prédiction de la trajectoire. Il comprend le calcul de : la probabilité qu'un POI(i) suivi d'un POI(j) soit observé dans la séquence d'apprentissage, la probabilité que cette série soit suivie d'un POI(t) observé et de la probabilité d'occurrence qu'un POI(i) suivi d'un POI(j) soit suivi par le même POI(t). Dans notre modèle nous considérons $l = 8$ états possibles $l =$ { D (directe), A (arrière), R (droit), L (gauche), AL (arrière gauche), PL (penché droit), AR (arrière droit), PR (penché droit) }. Les définitions pertinentes sont les suivantes :

Définition 1. *VHMM-(2,1) utilise $\lambda = (\pi, A_1, A_2, B_1, B_2)$ pour décrire le modèle. Supposons que l représente les états existants, POI représente la séquence d'état des points d'intérêts de la trajectoire, O représente la séquence d'états d'observation. VHMM-(2,1) est caractérisée par les notations suivantes :*

- $\pi = \{\pi\}$ *: la probabilité de l'état initial de la trajectoire, qui est connue dans notre cas.*

- $A_1 = \{A_{(j),k}\}$ *: La probabilité de transition d'un état (j) à un état (k).*
  $A_{((j),k)}$*=P(*$POI_{(k)}$*|* $POI_{(j)}$*) signifie la probabilité qu'il y ait un* $POI_{(j)}$ *suivi d'un* $POI_{(k)}$ *dans la séquence d'apprentissage (d'observation).*

- $A_2 = \{A_{(i)(j),k}\}$ *: La probabilité de transition à un état (k) après enchaînement des deux états [(i)(j)].*
  $A_{((i)(j),k)}$*=P(*$POI_{(k)}$*|* $POI_{(j)}$*,*$POI_{(i)}$*) signifie la probabilité qu'il y ait un* $POI_{(i)}$ *suivi d'un* $POI_{(j)}$ *enchaîné par un* $POI_{(k)}$ *dans la séquence d'apprentissage (d'observation).*

- $B_1 = \{B_k(l)\}$ *: La probabilité que l'on observe le symbole l, avec* $l = \{D, A, R, L, AL, PL, AR, PR\}$*, alors que le modèle se trouve dans l'état (k) dans un HMM-1 :* $B_k(l)$*=P(*$O_{(t)} = l|$ $POI_{(k)}$*).*

- $B_2 = \{B_k(l)\}$ *: La probabilité que l'on observe le symbole l, avec* $l = \{D, A, R, L, AL, PL, AR, PR\}$*, alors que le modèle se trouve dans l'état (k) dans un HMM-2 :* $B_k(l)$*=P(*$O_{(t)} = l|$ $POI_{(k)}$*).*

Maintenant, en se basant sur les probabilités de transitions A_1 et A_2 et les probabilité d'observation B_1 et B_2, nous déterminons le meilleur emplacement pouvant être le plus probable ($POI(k)$) dans le modèle HMM λ utilisant une séquence d'observation O : P($(POI(k)), O|\lambda$). Pour cela, en utilise l'algorithme de Viterbi [**?**] on définit dans Eq II.12 la variable intermédiaire $\Gamma_t(j)$ comme la probabilité du meilleur chemin amenant à l'état j à l'instant t :

$$\Gamma_t(j) = Max[P(j, O_t|\lambda)]$$

(II.12)

Et par récurrence, on calcule $\Gamma_{t+1}(k)$ définissant la probabilité maximale correspondant au meilleur $POI_{(k)}$ en utilisant **Définition 2** (cas HMM-2). Dans le cas où toutes les probabilités sont nulles, le serveur bascule en un HMM-1 et calcule la probabilité maximale correspondant au meilleur $POI_{(k)}$ en utilisant **Définition 3**.

Définition 2. *La probabilité de l'emplacement le plus prédit de l'instant* $(t+1)$ *correspondant à l'état k pour HMM-2 :*

$$\Gamma_{t+1}(k) = Max[\Gamma(j)A_{(i)(j),k}]|B_k(O_{t+1})$$

Définition 3. *La probabilité de l'emplacement le plus prédit de l'instant* $(t+1)$ *correspondant à l'état k pour HMM-1 :*

$$\Gamma_{t+1}(k) = Max[\Gamma(j)A_{(j),k}]|B_k(O_{t+1})$$

Nous adoptons cet algorithme de prédiction de configuration basé sur un modèle HMM d'ordre 2 (HMM-2) qui pourrait basculer vers un ordre 1 (HMM-1) lorsque le HMM-2 se trouve dans un contexte qui n'a jamais été expérimenté dans l'ensemble des données d'entraînement « Matching failed » (c-à-d : la probabilité qu'un POI(k) soit précédé des deux derniers POIs (POI(i), POI(j)) est nulle), l'obligeant à faire un tirage aléatoire du résultat, ce qui motive à définir un modèle HMM variable entre ordre 2 et ordre 1 « Variable HMM-(2,1) (VHMM-(2,1)) ». La Figure II.1 montre comment passer d'un ordre γ à un ordre $\gamma = \gamma - 1$ (dans notre cas d'un ordre 2 à un ordre 1).

Dans la Figure II.1, la première partie (Séquence de POIs) contient la séquence de γ POIs, la première recherche s'effectue sur une séquence d'un ordre γ élevé (Dans notre cas, γ=2). Si aucune correspondance n'est trouvée, l'ordre γ est réduit (γ=1). La recherche de correspondance se fait dans le bloc Séquences observées contenant toute la phase d'apprentissage. Dans ce bloc, les trajectoires d'apprentissage sont divisées en groupes où chaque groupe est composé de 3 POIs pour HMM-2 et 2 POIs pour HMM-1. Ensuite, le modèle de prédiction proposé est utilisé pour prédire le prochain POI en fonction de l'état actuel et des états précédents. Lorsqu'une correspondance est retrouvée, ce bloc ressort l'ordre du modèle HMM et la correspondance retrouvée (POI prédit). Sur la base du POI prédit et

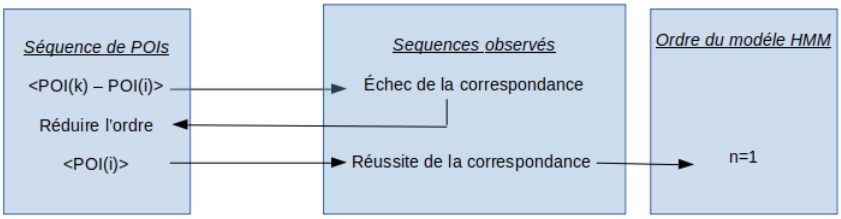


FIGURE II.1 – ordre γ du HMM

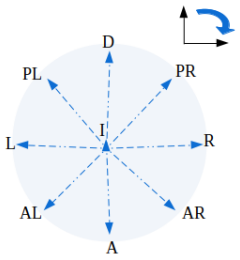


FIGURE II.2 – Les $l = 8$ POIs considérés

de l'emplacement des noeuds par rapport aux gateways, les configurations nécessaires sont optimisées et allouées de manière appropriée.

Comme mentionné précédemment, dans les différentes trajectoires en phase d'apprentissage ou en phase de test, nous considérons les POIs illustrés dans la Figure II.2. Le nombre de POIs différents est $l = 8$. Afin de valider le choix du VHMM-(2,1), nous proposons de le comparer à d'autres modèles HMM d'ordre fixe dans le cas de trajectoires connues.

3.2.2 Validation du modèle VHMM-(2,1)

Pour valider l'efficacité du modèle de prédiction de la mobilité VHMM-(2,1) par rapport aux modèles HMM simple (HMM-1, HMM-2, et HMM-3), nous considérons le critère de précision des prédictions. La précision des prédictions représente le rapport entre le nombre de prédictions correctes et le nombre total des positions d'une trajectoire connue et est calculée selon l'Eq (II.13).

$$Précision = \frac{Nombredeprédictionscorrectes}{Nombretotaldeprédiction} \tag{II.13}$$

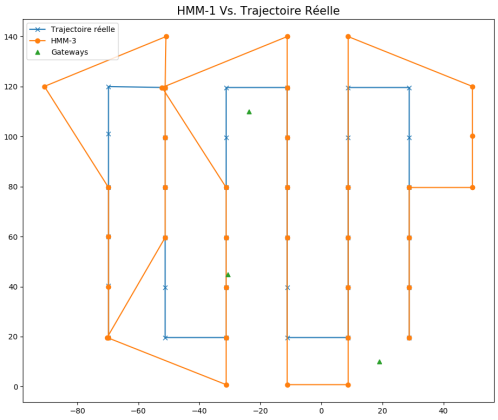
Nous évaluons les différents modèles de prédiction, HMM-1, HMM-2, HMM-3 et VHMM-(2,1) en prenant différents trajectoires connues. Nous commençons par définir 90 trajectoires qui seront utilisées dans la phase d'apprentissage ensuite nous expérimentons 5 noeuds empruntant des trajectoires différentes. Le noeud D1 emprunte une trajectoire en Zigzag. Il envoie des paquets de 10 Bytes toutes les deux minutes.

La Figure II.3 présente les trajectoires prédites comparées à la trajectoire Zigzag réelle traversée par D_1.

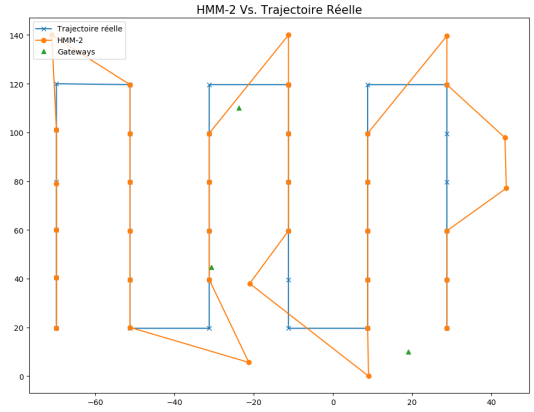
D'après cette représentation, nous constatons que la trajectoire prédite en utilisant VHMM-(2,1) se rapproche le plus de la trajectoire « Zigzag » réelle. Les résultats du modèle VHMM-(2,1) sont plus performants de ceux obtenus en utilisant les modèles HMM-1, HMM-2 et HMM-3. Par ailleurs, HMM-2 est meilleur que HMM-1 ou HMM-3.

L'expérimentation de 4 autres noeuds (D_2 à D_5) empruntant des trajectoires quelconques confirment la performance de HMM-(2,1) par rapport aux autres modèles. La Table II.2 présente le taux moyen de précision de la prédiction obtenu pour les 5 trajectoires testées avec les différents modèles de prédiction. Nous rappelons que les positions prédites sont comparées aux positions des trajectoires réelles.

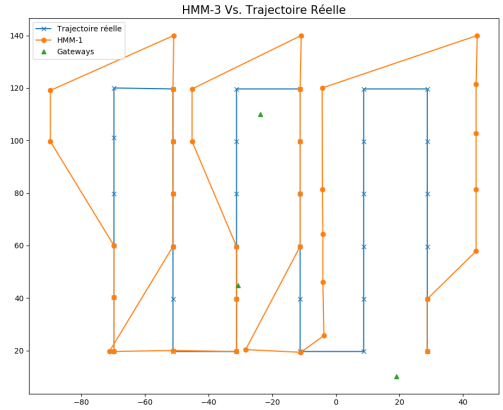
D'après les résultats obtenus dans la Table II.2, HMM-1 donne une précision moins précise que HMM-2 puisqu'il ne se base que sur le POI actuel pour prédire le POI suivant alors que



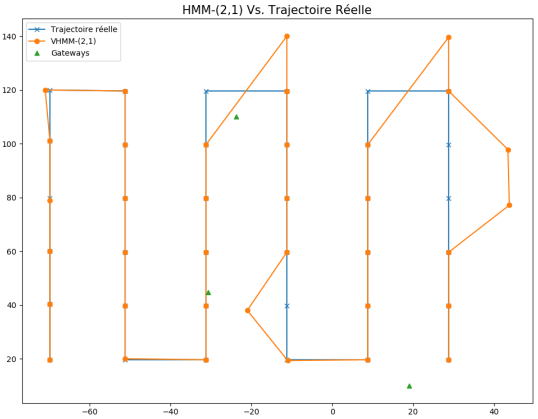
(a) Trajectoire réelle Vs. HMM-1



(b) Trajectoire réelle Vs. HMM-2



(c) Trajectoire réelle Vs. HMM-3



(d) Trajectoire réelle Vs. VHMM-(2,1)

FIGURE II.3 – Trajectoire en Zigzag Vs trajectoire prédite (HMM-1, HMM-2, HMM-3, VHMM-(2,1))

Table II.2. Précision de la prédiction

Node	VHMM-(2,1)	HMM-1	HMM-2	HMM-3
D_1	86.08%	54.87%	69.71%	42.25%
D_2	89.14%	57.12%	71.48%	47.51%
D_3	88.52%	55.58%	65.89%	44.47%
D_4	88.09%	52.38%	70.35%	48.02%
D_5	87.15%	54.24%	67.19%	47.87%

HMM-2 est basé sur 2 POIs historiques pour prédire la position suivante. En outre, le modèle HMM-3 est le moins efficace par rapport à ces deux derniers modèles, car il est basé sur l'historique de 3 POIs donc plus d'apprentissage. En effet, il a été montré dans [78] que, les prédicteurs du HMM au-delà du second ordre (γ=2) sont en général moins précis.

Par ailleurs, le modèle VHMM-(2,1) est plus performant que HMM-2 pour la prédiction de la mobilité dans le cas de déplacements irréguliers puisqu'il peut traiter les cas impossibles

dans HMM-2 en basculant vers HMM-1. D'où l'utilisation de ce modèle dans la suite de nos travaux pour la prédiction des déplacements des noeuds mobiles afin de leurs proposer les meilleures configurations.

3.2.3 Allocation de configuration

Une fois que le modèle VHMM-(2,1) détermine le prochain POI, le serveur calcule les coordonnées de cette position à travers les équations II.15, II.16 et II.17, en tenant compte du sens et de la direction du déplacement prédit (Figure II.2).

$$d_{parcour} = Vitesse \times P_{sending} \tag{II.14}$$

- Si déplacement vers D ou A :

$$X_k = X_j \pm d_{parcour} \tag{II.15}$$

- Si déplacement vers L ou R :

$$Y_k = Y_j \pm d_{parcour} \tag{II.16}$$

- Si déplacement vers PL, AL, PR et AR avec θ :angle de déplacement :

$$\begin{aligned} X_k &= X_j \pm (d_{parcour} \times \cos\theta) \\ Y_k &= Y_j \pm (d_{parcour} \times \sin\theta) \end{aligned} \tag{II.17}$$

Après estimation de la position, la distance entre le POI estimé et la gateway sera calculée et convertie en RSSI selon l'Eq II.9. Ensuite, à partir de l'estimation du RSSI, le taux de transition du mode a au mode b sera calculé en utilisant l'Eq II.10. Le taux le plus élevé correspondra au prochain mode b. Le serveur informe le noeud du nouveau mode de configuration à travers la commande $LinkADRReq$ (CID=$0x03$) [21].

4 – Conclusion

Dans ce chapitre, nous avons présenté les différentes versions améliorées d'ADR proposées dans la littérature ainsi que les principaux défis de l'ADR de base (ADR Basique). Nous avons ensuite proposé une solution (E-ADR) pour l'adaptation de débit pouvant s'adapter aux différentes conditions d'instabilité (Obstacles mobiles, noeuds mobiles, etc). Deux versions du E-ADR ont été proposées. La première est adaptée au cas de noeuds mobiles à trajectoire connue et se base sur l'algorithme de régression linéaire pour la prédiction de la prochaine configuration du noeud mobile. La seconde permet d'élargir la liste des applications IoT utilisant le protocole LoRaWAN et de résoudre la limitation de la version précédente aux trajectoires connues. Ceci, en s'adaptant aux contextes de trajectoires inconnues à travers un

modèle de prédiction basé sur le modèle de markov variable caché « VHMM-based E-ADR ». Ce modèle VHMM-(2,1) a été validé après une comparaison en termes de précision de la prédiction à d'autres modèles HMM (HMM-1, HMM-2, et HMM-3).

Dans le chapitre suivant, nous allons évaluer expérimentalement les performances des deux versions d'E-ADR et les comparer à ADR basique, ainsi que les variantes d'ADR les plus prometteuses de la littérature.

Chapitre III

Évaluation expérimentale d'E-ADR

Dans ce chapitre, une comparaison entre les variantes les plus pertinentes d'ADR et E-ADR est présentée. Le taux de perte (PLR), le temps de transmission (ToA), le délai et l'efficacité énergétique sont les principaux critères de comparaison. Nous considérons un scénario d'une application de ferme connectée pour l'évaluation des différents mécanismes. Cette comparaison sera faite pour un contexte fixe en présence d'obstacles mobiles dans un premier temps et ensuite pour un contexte mobile. En outre, les retransmissions seront autorisées pour les applications qui en auront besoin, ce qui nous permettra de voir l'impact de l'activation de ces retransmissions sur les différentes variantes. Nous utilisons les modules LoRa Waspmote SX1272 de libelium [49, 83] et STM32 de STMicroelectronics [84] pour évaluer les performances d'E-ADR. Bien entendu, nous avions implémenté en python les deux versions d'E-ADR, ainsi que les autres variantes de comparaison (ADR+, EXPLoRa-SF et EXPLoRa-AT).

1 – Conditions d'évaluation

Étant une application motivante pour l'utilisation de LoRaWAN, nous considérons l'émulation d'un scénario dans une ferme connectée « Smart farming » [85]. avons utilisé dans les différents scénarios 5 noeuds LoRa (Waspmote SX1272 et STM-32 Discovery) ainsi que 3 gateways Waspmote SX1272. Les tests sont effectués en outdoor. La surface d'évaluation et l'emplacement des noeuds LoRa sont présentés dans chaque scénario. Dans les différents tests qui seront présentés, les noeuds utiliseront les 10 configurations prédéfinies par Waspmote avec une puissance de transmission pour tous les nœuds fixée à 14dBm et le CR à 4/5 (Table II.1).

Les variantes d'ADR évaluées ont souvent des fréquences de mise à jour de la configuration différentes (par exemple tous les 3 paquets dans E-ADR, et 20 paquets dans ADR Basique). Afin de comparer les différentes variantes de manière équitable en se concentrant uniquement sur l'efficacité algorithmique des différentes stratégies sans l'influence ni de la fréquence d'exécution de l'algorithme ni des retransmissions, nous exécutons la mise à jour de la configuration tous les $n = 3$ paquets pour toutes les variantes ADR dans le cas de trajectoires connues. Les 3 premières transmissions utilisent SF12 pour toutes les variantes afin de garantir la réception des premiers paquets par les gateways. Le processus d'allocation commence juste après la réception de ces $n = 3$ paquets. Dans le cas de la version « VHMM-based E-ADR »,

la mise à jour de configuration est exécutée après chaque réception de paquet ($n = 1$) puisque la trajectoire est irrégulière.

De plus, puisque nos scénarios consistent en un nombre limité de noeuds par rapport au nombre de SF et des canaux de transmission (3 gateways), nous faisons face rarement au problème de perte de paquets causée par une interférence (c-à-d. lorsque les 5 noeuds utilisent le même SF, le même canal et transmettent tous au même temps). En outre, comme spécifié par LoRaWAN, les retransmissions peuvent être autorisées ou pas à la demande. En pratique, ce choix est basé sur l'importance de la donnée à transmettre (le type d'application). Par exemple, dans le cas d'une application de sécurité, les messages envoyés sont souvent de type confirmé (retransmissions autorisées). Nous nous intéressons dans notre étude aussi bien au trafic confirmé qu'au trafic non confirmé. En effet, il est important de voir l'effet des retransmissions sur la consommation du temps de service (Duty cycle limité à 1%) autorisé pour chaque noeud malgré son efficacité pour augmenter le nombre de paquets reçus. Notons que, dans nos expérimentations, nous considérons comme "paquets perdus" les paquets qui n'ont pas pu être transmis à cause du dépassement de Duty Cycle C_n et que ça sera inutile de les envoyer durant le prochain cycle C_{n+1}.

Dans le cas des applications nécessitant la transmission de messages confirmés (retransmissions activées), nous supposons qu'un paquet i envoyé par un noeud D_j peut être retransmis au maximum $\alpha_{i,j}$ fois tant qu'il n'est toujours pas acquitté. Nous définissons $\alpha_{i,j}$ dans l'Eq (III.1) comme étant le nombre maximal de retransmissions possibles du paquet i (envoyé par D_j) avant l'envoi du prochain paquet programmé $(i + 1)$. Dans l'Eq (III.1), $T_{delay}(i, j)$ correspond au délai de transmission du paquet i envoyé par D_j et représente le ToA de tout le paquet ainsi que les deux fenêtres de réception de l'acquittement (ACK), et $P_{sending}(i, j)$ représente la période d'envoi correspondant à la longueur de l'intervalle [T_i,T_{i+1}].

$$\alpha_{i,j} = \frac{P_{sending}(j)}{T_{delay}(i, j)} \tag{III.1}$$

Durant ces $\alpha_{i,j}$ retransmissions, si le noeud ne reçoit pas d'acquittement après le seuil $ADRACKLIMIT$ ($ADRACKLIMIT < \alpha_{i,j}$), le paquet est considéré perdu (mauvaise allocation de SF) et le noeud incrémentera son SF (SF=SF+1) pour augmenter la couverture ainsi la probabilité d'atteindre la gateway et tente de retransmettre le paquet. Par ailleurs, lorsque les $\alpha_{i,j}$ retransmissions sont atteintes, le paquet ne sera plus retransmis et sera considéré perdu. Dans [3, 21], le seuil $ADRACKLIMIT$ est fixé à 64 par défaut.

Le calcul de l'énergie [86] est basé sur les courants des 6 états (T_x, $1^{ère}$ attente, la $1^{ère}$ R_X, $2^{ème}$ attente, la $2^{ème}$ R_X et l'état de veille) (Eq III.2) pour le cas de message confirmé et de 2 états pour un message non confirmé (T_x et l'état de veille) (équation III.3). La figure III.1 décrit ces différents états impliqués dans le cas de retransmissions autorisées. Tout d'abord, le nœud est en état de transmission (T_x). Après la transmission, le noeud désactive l'activité radio et attend (état 2) jusqu'à ce qu'il mette la radio en mode réception (T_{w1w}) et reste dans le même état pendant la durée de la première fenêtre de réception T_{Rx1w} (état 3). Comme aucun préambule entrant n'est détecté, la première fenêtre de réception est fermée, et le nœud attend T_{w2w} (état 4) jusqu'au début de la deuxième fenêtre de réception T_{Rx2w} (état 5). Pendant ce dernier, la radio du nœud est activée pour d'éventuelles données entrantes, jusqu'à ce que la deuxième fenêtre de réception soit fermée. Après cela, l'interface radio est

désactivée T_{Sleep} (état 6). Nous tenons à noter que si un préambule est détecté pendant la première fenêtre de réception (état 3), le noeud passe directement à l'état 6 sans ouvrir la deuxième fenêtre de réception (les états 4 et 5 sont ignorés).

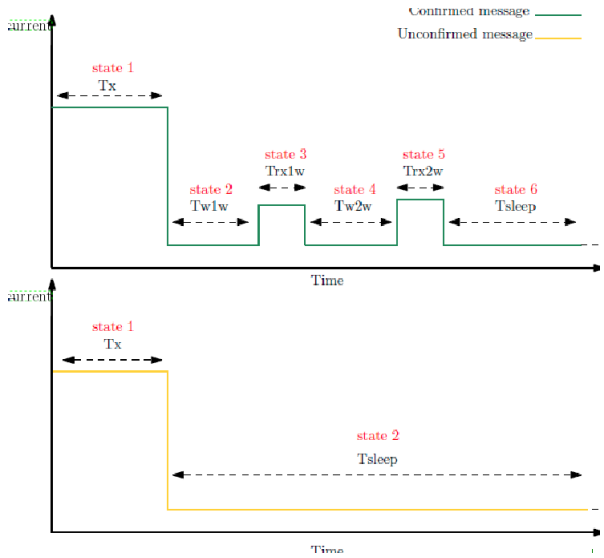


FIGURE III.1 – Courant d'alimentation pour le nœud SX1272 LoRa dans différents États à $U = 3V$

La Table III.1 indique le courant consommé par le nœud LoRa SX1272 dans les 6 états à une tension de $U = 3$ volts, comme indiqué dans [87] .

$$E_{confirmé} = (T_{Tx} \times I_{Tx} + T_{w1w} \times I_{w1w} + T_{Rx1w} \times I_{Rx1w} + T_{w2w} \times I_{w2w} + T_{Rx2w} \times I_{Rx2w} + T_{Sleep} \times I_{Sleep}) \times U \quad \text{(III.2)}$$

$$E_{non-confirmé} = (T_{Tx} \times I_{Tx} + T_{Sleep} \times I_{Sleep}) \times U \quad \text{(III.3)}$$

Table III.1. Courant d'alimentation pour le nœud SX1272 LoRa pour les différents États à $U = 3Volts$

État	symbole	valeur
veille	I_{Sleep}	0.1 µA
1^{st} attente	I_{w1w}	1.5 µA
2^{nd} attente	I_{w2w}	1.5 µA
R_{X1}	I_{Rx1w}	11.2 mA
R_{X2}	I_{Rx2w}	11.2 mA
T_x	I_{Tx}	28 mA

Dans ce qui suit, nous proposons d'évaluer la version E-ADR dans un contexte fixe en présence d'obstacles mobiles afin de prouver l'efficacité de notre approche par rapport aux autres variantes. Nous l'évaluons par la suite dans un contexte mobile afin de valider son

efficacité dans des contextes plus instables (mobilité). Enfin, nous présenterons l'évaluation de performance de la version « VHMM-based E-ADR » dans un contexte mobile à trajectoires inconnues. Parmi les variantes ADR, nous avons choisi d'évaluer et de comparer E-ADR à : ADR Basique [3], ADR+ [4], EXPLoRA-SF [5], EXPLoRA-AT [5]. Nous avons implémenté tous ces mécanismes dans le but de les évaluer et de faire une analyse dans les mêmes conditions, en se basant sur les métriques suivantes : le taux de perte de paquets, le temps de transmission consommé (ToA), le délai de transmission et la consommation énergétique (E).

2 – Évaluation de E-ADR dans le cas de noeuds statiques et des obstacles mobiles

Dans cette section, nous voulons évaluer les performances du E-ADR et son efficacité dans le contexte d'un réseau de noeuds statiques où des obstacles mobiles sporadiques peuvent s'introduire entre le noeud et les gateways et nuire à la transmission des paquets (tels que les mouvements des arbres et des plantations dans les parcelles lors d'une tempête de vent, le passage d'un troupeau d'animaux ou un grand tracteur devant un capteur, etc.). Le but de cette évaluation est de montrer que l'E-ADR résiste aux événements pouvant dégrader la QoS du réseau LoRaWAN.

2.1 Scénario de test

Nous considérons 5 noeuds fixes qui transmettent à une fréquence $P_{sending(j)}$ pour chaque D_j.

Les gateways sont fixes et leurs positions sont les suivantes : G_a = [19, 10.07], G_b = [-30.71, 44.8] et G_c = [-23.74, 110.08]. Ci-dessous, les applications utilisées par les différents noeuds :

- Le premier noeud D_1 est un capteur pour plantes mis dans une serre qui recueille différents signaux tels que l'état de l'hydratation de la plante, l'état de sa croissance, la teneur en engrais et les nutriments de la plante. D_1 envoie 8 paquets de 30 Bytes chacun ($P_{sending(1)}$=7 min).
- Le deuxième noeud D_2 concerne un silo de stockage d'aliments pour les animaux. Ce capteur envoie 12 paquets de 15 Bytes chacun toutes les 5 minutes ($P_{sending(2)}$=5 min) pour informer sur la quantité des graines restante dans le silo et la quantité à rajouter lorsque les animaux se nourrissent ainsi le degré d'humidité du silo afin d'éviter les infestations et les risques d'échauffement.
- Les noeuds D_3 et D_4 sont deux capteurs de mesure de maturité de deux arbres à fruits. Ils fournissent à l'agriculteur un diagnostic sur l'état des arbres fruités, en envoyant 6 paquets de 20 Bytes chacun, toutes les 10 min. La différence entre les deux capteurs se situent dans leurs emplacements par rapport aux gateways (D_3 se trouve plus proche des gateways que D_4, voir Fig III.2).
- Le dernier noeud D_5 est un capteur pour l'identification des animaux présents dans les stabulations en envoyant 100 Bytes chaque $P_{sending(5)}$=5 min.

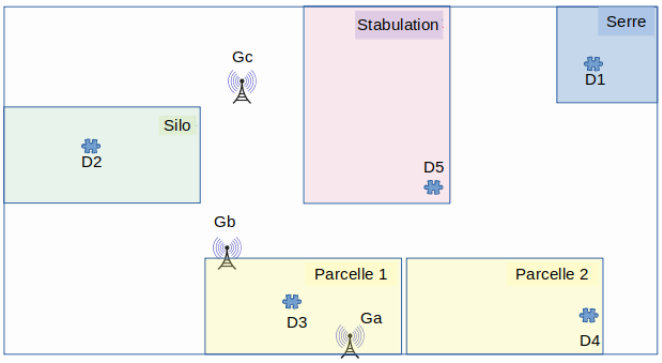


FIGURE III.2 – Position des gateways et des noeuds - Scénario statique

Nous supposons que tous les noeuds envoient durant le même cycle et que ces derniers sont dédiés à des applications ne nécessitant pas l'envoi de message confirmé (Pas de retransmissions). Les noeuds D_1, D_3 et D_4, étant mis dans des serres et des parcelles, peuvent se trouver devant des obstacles tels que les arbres et les grandes plantes lors de la présence de mauvaises conditions météorologiques ou la présence des grands tracteurs. Les noeuds D_2 et D_5 sont souvent faces à un nombre important de troupeaux d'animaux bloquant et influençant momentanément le signal émis par les capteurs. Dans nos expérimentations, nous créons artificiellement des obstacles entre les noeuds et les gateways afin d'évaluer leur influence sur la performance du réseau et leur impact sur les variantes ADR. Ces obstacles se présentent lors de l'émission des noeuds, en se positionnant entre les noeuds et les gateways dans les différents angles (tournant autour des noeuds).

2.2 Évaluation de performance

Nous évaluons E-ADR et le comparons aux différentes variantes. Nous exécutons la mise à jour de configuration tous les $n = 3$ paquets reçus et seulement une seule fenêtre de liaison descendante sera ouverte après chaque envoi de $n = 3$ paquets afin de recevoir l'acquittement de configuration ACK_{Config}. Tous les noeuds commencent leurs 3 premières transmissions en utilisant $mode1$ (SF12). E-ADR sera évalué et comparé aux autre variantes ADR en termes de PLR, ToA et consommation énergétique.

2.2.1 Évaluation du PLR

Dans nos expérimentations, le paquet peut être perdu soit à cause du mauvais SF alloué, des interférences, ou le dépassement de la limite du Duty Cycle dans certains cas. Le taux de perte de paquets (PLR) d'un noeud D_j est défini comme étant le nombre de paquets perdus (dû au mauvais SF) et les paquets non transmis (dû au dépassement du Duty Cycle) sur le nombre total des paquets programmés à être transmis par D_j.

Dans cette partie, nous évaluons et discutons le PLR des 5 noeuds pour E-ADR et les variantes ADR. Les résultats obtenus sont présentés dans la Table III.2. De plus, les Figures III.3- III.7 détaillent l'évolution du PLR dans le temps pour les différents noeuds utilisant les différentes variantes.

Table III.2. Le PLR des noeuds

Noeud	ADR Basique	ADR+	EXPLoRA-SF	EXPLoRA-AT	E-ADR
D_1	37.5%	25%	37.5%	0%	0%
D_2	41.66%	25%	25%	41.66%	0%
D_3	33.33%	0%	50%	0%	0%
D_4	33.33%	33.33%	0%	0%	0%
D_5	25%	8.33%	16.66%	75%	0%

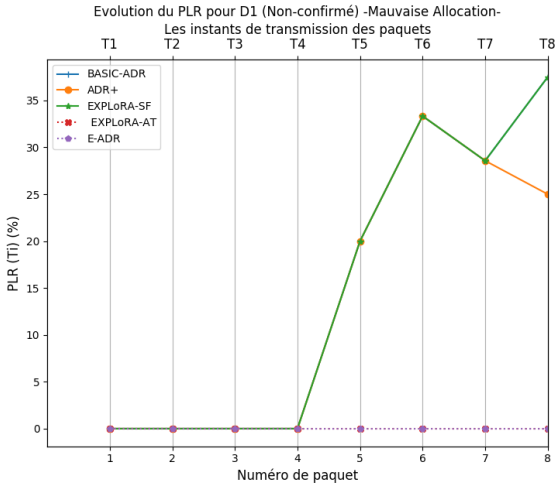


FIGURE III.3 – Évolution du PLR pour D_1

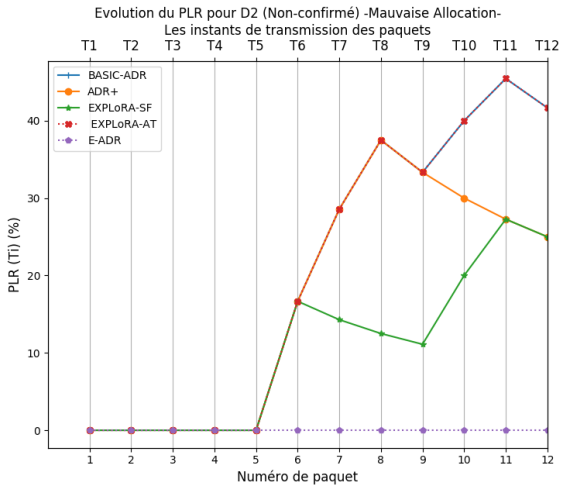


FIGURE III.4 – Évolution du PLR pour D_2

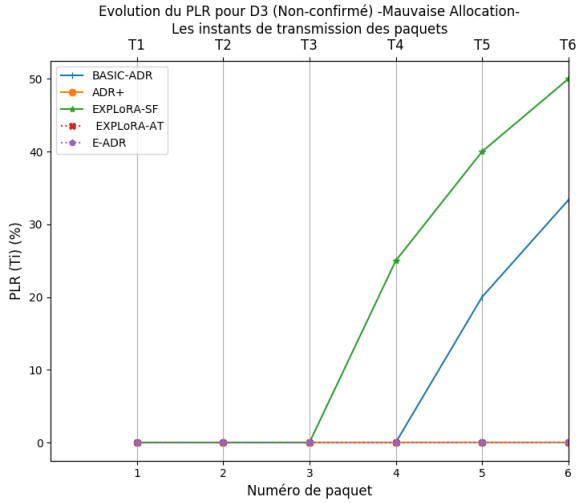


FIGURE III.5 – Évolution du PLR pour D_3

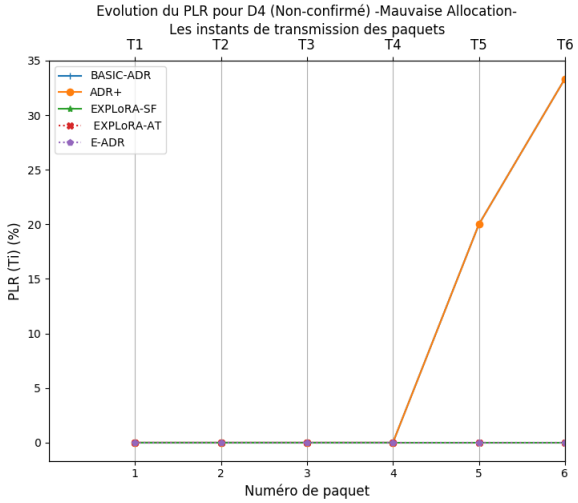


FIGURE III.6 – Évolution du PLR pour D_4

Nous constatons d'après les Figures III.3- III.7 que tous les noeuds ne notent pas de perte des premiers paquets puisqu'ils utilisent tous SF12 et n'envoient pas au même rythme

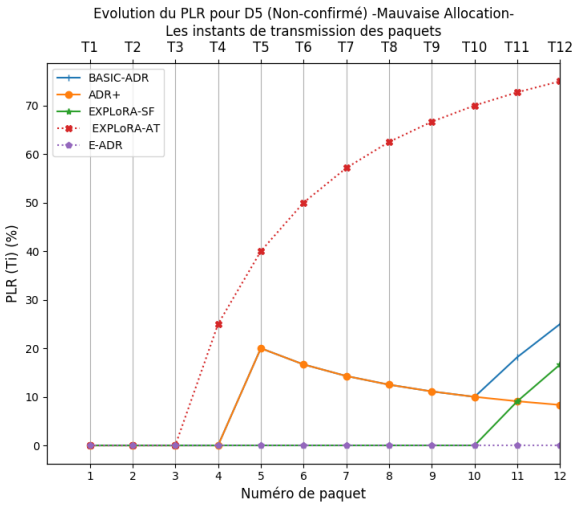


FIGURE III.7 – Évolution du PLR pour D_5

(fréquence d'envoi). Un peu plus tard (à partir de l'envoi du $3^{ème}$, $4^{ème}$ ou $5^{ème}$ paquet, selon la stratégie et le noeud), on note des pertes à cause d'une configuration non adaptée en présence d'obstacles.

Par exemple, pour le cas du noeud D_1, le paquet numéro 8 (le dernier) est perdu en utilisant l'EXPLoRA-SF et ADR Basique (croissance de la courbe) et est reçu en utilisant E-ADR, ADR+ et EXPLoRA-AT. La perte du $8^{ème}$ paquet en utilisant EXPLoRA-SF et ADR Basique est due au fait qu'un obstacle s'est présenté à T_8 et que le SF alloué n'était pas suffisant pour y faire face.

En résumé, toute croissance dans les courbes signifie la présence d'un obstacle à cet instant induisant une perte du paquet à cause d'une allocation de SF insuffisante. Par conséquent, la décroissance des courbes (Ou leurs stabilité à 0, par exemple pour D_4 où EXPLoRA-SF et EXPLoRA-AT résultent en un PLR nul ne signifie pas une meilleure performance, puisque ces stratégies sont basées sur la classifications des SF selon les valeurs RSSI de tous les noeuds (EXPLoRA-SF) et leurs classification selon la quantité de données (EXPLoRA-AT), ce qui veut dire que le SF attribué pourrait dans certains cas éviter la perte de paquet, mais aux dépens d'une sur-consommation énergétique qui sera évaluée dans la section suivante.

D'autre part, le fait que D_5 souffre d'un PLR important en utilisant EXPLoRA-AT, comparant aux PLRs des autres noeuds, revient au principe de la stratégie d'allocation où le noeud envoyant une grande quantité de données se voit attribué un faible SF, ce qui est le cas de D_5. Ce qui explique le grand PLR présenté dans la Table III.2 pour le cas de D_5 et la croissance de la courbe (en rouge) tout au long de l'envoi des paquets dans la Figure III.7.

De plus, nous constatons dans la Table III.2 que l'ADR+ peut mieux supporter la présence d'obstacles dans certains cas par rapport aux autres variantes (Par exemple, (D_3). Ceci revient au fait que l'ADR+ se base sur la moyenne des valeurs SNRs, correspondant à la valeur moyenne des RSSIs reçus (en d'autres termes), ce qui lui permet, dans le cas où les valeurs faibles (présence d'obstacles) sont majoritaires par rapport aux valeurs élevées dans la liste des n valeurs reçues, d'allouer un SF élevé pouvant faire face aux obstacles.

Les différents résultats montrent qu'en utilisant l'E-ADR, les différents noeuds reçoivent

tous leurs paquets, ceci revient au fait que la stratégie E-ADR proposée tient compte de la variation des n valeurs RSSI reçues, c'est-à-dire, toute variation des conditions est prise en considération dans la prédiction de la prochaine configuration du noeud.

Dans la prochaine section, nous évaluons le ToA et donc la consommation énergétique afin de confirmer que les stratégies, à l'exception du E-ADR, bénéficient d'un faible PLR au prix d'une sur-consommation énergétique dans certains cas.

2.2.2 Évaluation du ToA et de la consommation énergétique

Le temps de transmission (ToA) ainsi que la consommation énergétique sont aussi des critères importants dans l'évaluation de performance de la QoS dans le protocole LoRaWAN.

Les Figures III.8- III.12 présentent le ToA consommé par chaque paquet transmis par chaque noeud utilisant les différentes stratégies ADR y compris E-ADR. La Table III.3 présente la consommation énergétique des différents noeuds utilisant les différents ADR.

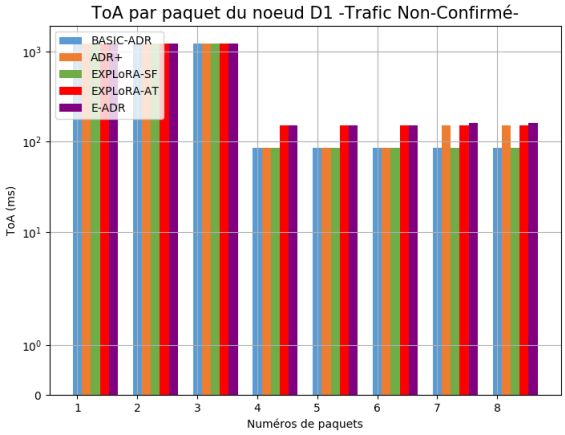


FIGURE III.8 – Évaluation du ToA pour D_1

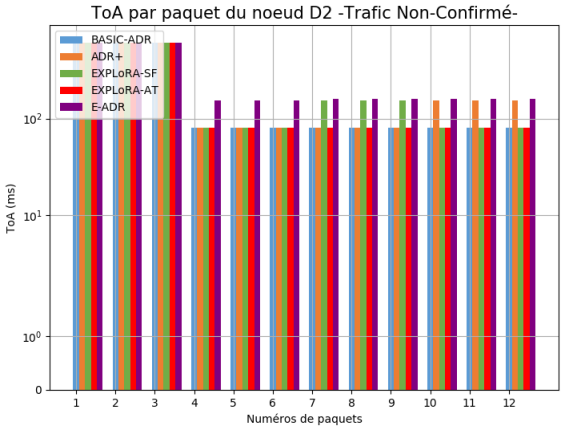


FIGURE III.9 – Évaluation du ToA pour D_2

Table III.3. La consommation énergétique des noeuds (J)

Noeud	ADR Basique	ADR+	EXPLoRA-SF	EXPLoRA-AT	E-ADR
D_1	3.11	3.16	3.11	3.22	3.23
D_2	3.05	3.12	3.12	3.05	3.32
D_3	2.44	2.47	2.43	2.61	2.47
D_4	2.58	2.58	2.65	2.65	2.65
D_5	8.15	8.57	8.57	5.72	9

Pour la transmission des 3 premiers paquets, tous les noeuds utilisent SF12. D'où l'égalité du ToA de ces paquets pour toutes les stratégies ADR pour tous les noeuds. Après la réception des premiers paquets, les noeuds allouent une nouvelle configuration selon la stratégie ADR utilisée, et c'est à partir de là que la différence entre les stratégies en termes de ToA est observée. Nous rappelons que selon les SNRs/RSSI reçus, les stratégies ADR Basique,

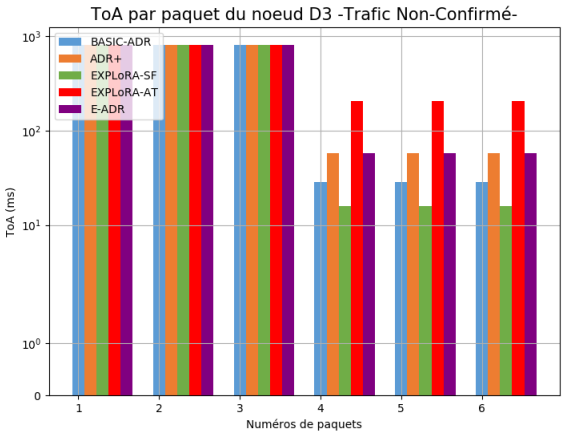


FIGURE III.10 – Évaluation du ToA pour D_3

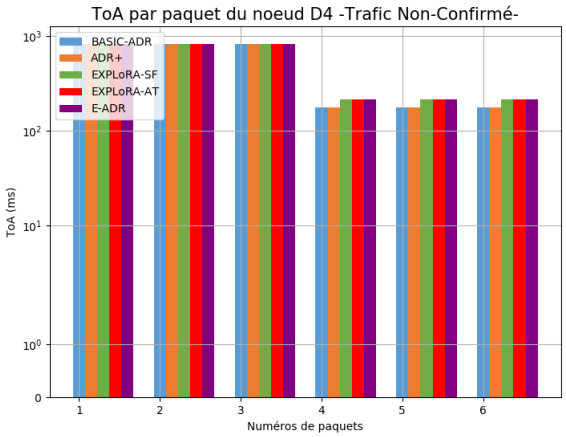


FIGURE III.11 – Évaluation du ToA pour D_4

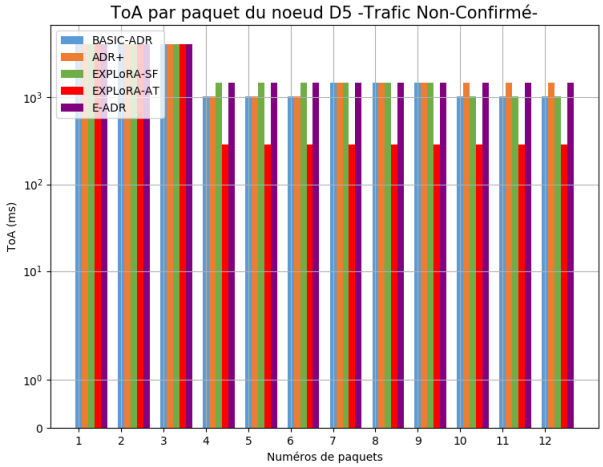


FIGURE III.12 – Évaluation du ToA pour D_5

ADR+, EXPLoRA-SF effectuent une allocation de configuration selon les SNR/ RSSI reçus. EXPLoRA-AT effectue son allocation selon la quantité de données que les différents noeuds doivent envoyer. E-ADR calcule après chaque $n = 3$ paquets reçus la variation de la position du noeud selon les RSSI reçus et estime les $n = 3$ prochains RSSIs selon cette variation. Normalement, pour les noeuds fixes la variation qu'on a appelé précédemment dans la section 3 « $Avg_{variation}$ » devrait être nulle sauf lors de la présence d'obstacles. Donc, dans un cas de noeuds fixes, la variation du RSSI permet à E-ADR de comprendre la présence d'obstacles et de préméditer la configuration par rapport aux autres stratégies.

Nous remarquons dans les Figures III.8- III.12 que pour certains paquets, les variantes ADR consomment moins de ToA par rapport à E-ADR. Ceci revient au fait que ces stratégies ayant alloué un faible SF, ne considèrent pas les variations de SNR/RSSI. Ce qui fait que plusieurs paquets sont souvent perdus à cause d'une allocation d'un SF non-adéquat (sous-optimal). Par ailleurs, E-ADR présente dans la plupart des cas un ToA supérieur puisqu'il assure la transmission de tous les paquets en offrant un débit faible pour faire face aux

obstacles.

Inversement, EXPLoRA-AT résulte en un ToA plus élevé que celui du E-ADR dans le cas de D_3 malgré qu'elle n'a pas noté non plus de pertes. Ceci est dû forcément à une sur-allocation de SF (SF plus grand à l'optimal) induisant à une sur-consommation énergétique.

En résumé, les stratégies qui présentent plus de perte (variantes autres que E-ADR) vont noter un ToA plus faible et donc une consommation énergétique plus faible. Par ailleurs dans la cas où les autres variantes présentent un PLR nul, nous remarquons que E-ADR présente en général une consommation énergétique plus faible ou égale.

Pour synthétiser, selon les tests effectués dans le cas de noeuds statiques en présence d'obstacles, nous concluons qu'E-ADR surpasse les variantes ADR existantes dans la littérature en termes de PLR au prix d'une légère augmentation du ToA et de la consommation énergétique. Nos résultats expérimentaux montrent que bien que ADR+, l'EXPLoRA-SF et EXPLoRA-AT surpassent l'ADR Basique, E-ADR est meilleur en termes de taux de succès des transmissions en s'adaptant aux cas d'instabilités des environnements. Dans ce qui suit, nous allons évaluer l'approche E-ADR dans un contexte de noeuds mobiles.

3 – Évaluation d'E-ADR dans le cas de noeuds mobiles à trajectoires connues

Dans cette section, nous présentons les résultats de l'évaluation des performances du E-ADR en le comparant aux variantes ADR proposées dans un contexte de noeuds mobiles avec trajectoires connues. Nous considérons l'émulation d'un scénario pour la même application de ferme connectée « Smart farming » [85], où 5 noeuds (drones et robots) sont manipulés et déplacés manuellement (pour assurer la reproductibilité de tous les tests) dans un champs ouvert en plein air (outdoor). Ces 5 noeuds se déplacent en même temps utilisant différents intervalles d'envoi de paquets ($P_{sending}(j)$) (j est le numéro du noeud) et vitesses de déplacement. Les résultats des mesures rapportés représentent la moyenne de 11 tests.

3.1 Scénario de test

Les 5 nœuds LoRa empruntent différents modèles de mobilité, couvrant une zone : S=($[X_{min}, X_{max}] = [-70, 40]$, $[Y_{min}, Y_{max}] = [10, 100]$). Contrairement au premier scénario, les nœuds transmettent leurs données pendant plusieurs cycles d'une heure (C_n= 1 heure, où n est le numéro de cycle) avec une limitation du Duty Cycle à 36sec/C_n.

La Figure III.13 présente les 3 gateways ayant respectivement les positions suivantes : G_a = [19, 10.07], G_b = [-30.71, 44.8] and G_c = [-23.74, 110.08]. Nous supposons également que les nœuds D_1, D_2, D_3, D_4 et D_5 présentés dans la Figure III.13 utilisent les différents modèles de mobilité présentés dans les Figures (III.14- III.16) où chaque point correspond à l'instant de transmission d'un paquet (qui peut être segmenté en l trames LoRa selon la taille de paquet et du mode de configuration utilisé).

Les nœuds sont dédiés aux différentes applications de ferme intelligente :

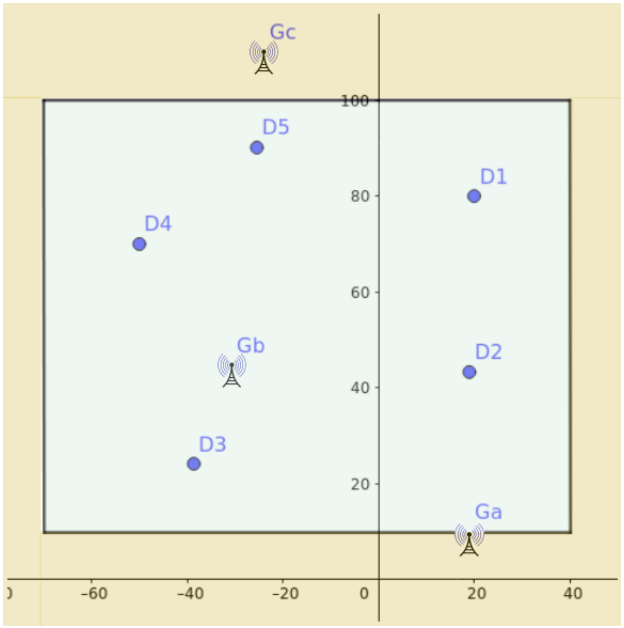


FIGURE III.13 – Position des gateways et des noeuds

— L'agriculteur doit disposer d'un rapport complet de l'état de la superficie d'une parcelle donnée dans la ferme. Ainsi, le nœud D_1 est un robot de désherbage des champs qui envoie 120 paquets de 25 octets chacun pendant 10 cycles (10C) (c-à-d. $P_{sending}(i)$=5min, ou $1pkt/5min$). Il couvre la superficie S en utilisant le modèle de mobilité en « Zigzag » à une vitesse de $2m/min$.

— Les parcelles aux rangées très hautes et denses sont difficiles à accéder à pied ou par des robots sans endommager les récoltes. Ainsi, le noeud D_2 est un drone (en supposant qu'il ait une capacité de récupération d'énergie) qui vole au-dessus de S en utilisant le modèle de mobilité « Zigzag » à une vitesse de $4m/min$. Il envoie 60 paquets de 25 octets pendant 5 cycles (5C) ($1pkt/5min$).

— Pour la plantation de la salade, l'agriculteur installe un robot qui génère une carte et un modèle en 3D de la zone correspondante pour la plantation de la salade. Ainsi, le nœud D_3 est un robot Vision qui aide à la plantation de laitue dans les zones appropriées. Il couvre S en suivant un modèle de mobilité « carré réduit » à une vitesse de $5m/min$. Il envoie 57 paquets de 25 octets pendant 4 cycles (4C) ($1pkt/4min$).

— Un autre drone (en supposant qu'il ait une capacité de récupération d'énergie) est conçu pour la distribution précise de pesticides, d'engrais et d'herbicides liquides. Le nœud D_4 traverse S à une vitesse de $5m/min$, suivant un modèle de mobilité « carré réduit », et envoie 29 paquets de 25 octets pendant 8 cycles (8C) ($1pkt/15min$).

— Le nœud D_5 est un robot pour l'irrigation. Il identifie les champs arides et secs, informe l'agriculteur sur ces zones et calcule la quantité d'eau nécessaire pour les humidifier. Le

nœud D_5 se déplace autour de la zone de S à une vitesse de $2m/min$, en suivant un modèle de mobilité « carré simple », et en envoyant 20 paquets de 250 octets pendant 4 cycles (4C) ($1pkt/12min$).

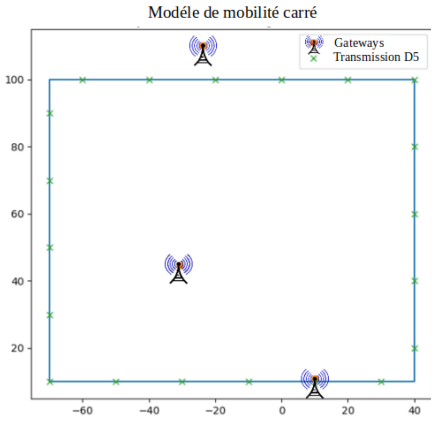


FIGURE III.14 – modèle de mobilité « carré » (D_5)

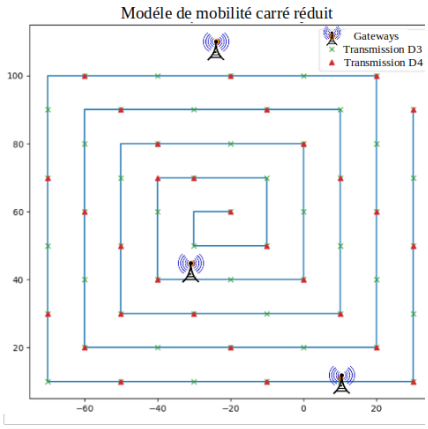


FIGURE III.15 – modèle de mobilité « carré réduit » (D_3,D_4)

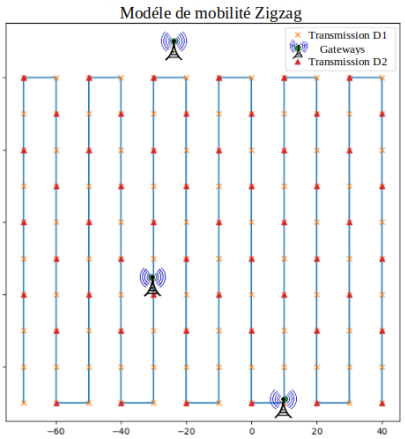


FIGURE III.16 – modèle de mobilité « Zigzag » (D_1,D_2)

Dans ce qui suit, nous évaluons les performances du réseau en termes de taux de perte (PLR), ToA et la consommation énergétique correspondant selon [86], et aussi le délai aller-retour (RTD).

Les applications émulées dans ce scénario peuvent nécessiter des acquittements à leurs transmissions afin d'assurer une bonne supervision des champs et des parcelles de la ferme connectées. Dans ce cas, nous proposons de tester le scénario dans un cas de transmissions confirmées (avec acquittements) et dans un cas de transmission non-confirmées (sans acquittements). Dans le premier cas, nous nous intéressons à l'analyse de l'effet de l'activation des retransmissions sur le PLR et le ToA en variant le nombre limite de retransmissions $ADRACKLIMIT$ avant changement de la configuration (incrémentation de SF). (Voir Chapitre 2).

3.2 Cas 1 : Utilisation des messages confirmés - retransmissions autorisées -

Chaque noeud définit un compteur de retransmission pour chaque paquet émis jusqu'à sa transmission. Ce compteur a pour limite ADRACKLIMIT. Si un paquet n'est pas acquitté avant d'atteindre cette limite alors une augmentation du SF sera nécessaire. Dans la suite, nous présentons les résultats des différents tests effectués.

3.2.1 Évaluation du PLR

Le taux de perte de paquets (PLR) est un critère de performance important puisque d'une part, il détermine la fiabilité de la communication, d'autre part son augmentation entraîne

un grand nombre de retransmissions, qui à son tour augmente le ToA ainsi que l'énergie. De plus, l'augmentation du ToA induit à un dépassement de la limite du Duty Cycle (36 sec par heure), et donc de nouvelles pertes. Le PLR d'un noeud D_j est défini comme étant le nombre de paquets perdus (dû au mauvais SF) plus les paquets non transmis (dû au dépassement du Duty Cycle) sur le nombre totale des paquets programmés à être transmis par D_j. Notons que dans le premier cas, un paquet est considéré perdu si il n'est pas acquitté après $\alpha_{i,j}$ retransmissions.

La Table III.4 présente le PLR enregistré par les noeuds D_j. Nous observons un PLR important dans tous les cas ADR à part E-ADR qui est expliqué par le fait que les SFs alloués par les variantes ADR sont très petits et que les valeurs optimales ne sont pas atteintes malgré l'incrémentation des SFs à chaque $ADRACKLIMIT$.

Table III.4. Le PLR des noeuds

Node	ADR Basique	ADR+	EXPLoRA-SF	EXPLoRA-AT	E-ADR
D_1	57.5%	43%	47%	80.78%	0%
D_2	81.67%	60%	60%	0%	0%
D_3	19.28%	10.52%	8.77%	0%	0%
D_4	27.58%	27.58%	27.58%	0%	0%
D_5	35%	35%	25%	20%	0%

Nous observons qu'en utilisant l'ADR+, les noeuds D_1, D_2, et D_3 résultent en un faible PLR comparé à la variante ADR basique, ceci revient au fait que l'ADR+ permet la considération de la variation de la force du signal (RSSI) dans certains cas et ceci en se basant sur la moyenne des valeurs RSSIs reçues au lieu de la valeur maximale.

EXPLoRA-SF, basée sur la distribution des SFs sur le nombre de noeuds disponibles dans le réseau, permet soit un SF assez grand (ToA élevé, pas de retransmissions) ou un SF assez faible (grand PLR, beaucoup de retransmissions). Ceci est dû au fait que EXPLoRA-SF distribue les SFs sans tenir compte des limites RSSI.

Dans le cas de EXPLoRA-AT, les noeuds transmettant une petite quantité de donnée comme D_2, D_3 et D_4 bénéficient d'un petit PLR, puisque l'allocation d'un grand SF permet la transmission de tous les paquets correctement. D_1 et D_5 transmettant une grande quantité de données (et s'éloignant des gateways) sont alloués de faibles SFs induisant à un PLR élevé. Pour tous les noeuds, E-ADR ne fournit aucune perte (PLR=0%).

En outre, le PLR pourrait dépendre aussi de la fréquence d'envoi et de la vitesse des noeuds. D_1 et D_2 traversent la même trajectoire « zigzag », cependant la vitesse est différente. Nous remarquons que le PLR de D_2 est supérieur à celui de D_1 en raison de l'augmentation de la vitesse du noeud D_2. En effet, lorsque la vitesse du noeud est plus importante, l'emplacement du noeud change rapidement et donc la variation du RSSI est plus importante d'une transmission à une autre.

De même, D_3 et D_4 traversent la même trajectoire à la même vitesse, mais cette fois-ci les fréquences d'envoi de paquets $P_{sending}$ sont différentes. L'augmentation de la fréquence d'envoi de D_3 (1pkt/4min) par rapport à D_4 (1pkt/15min) impacte sur le PLR. En effet, lorsque la fréquence d'envoi est faible (D_4), le RSSI varie rapidement entre deux transmissions, induisant ainsi un PLR élevé.

En ce qui concerne EXPLoRA-AT, d'une part, pour un même $P_{sending}$ et une même trajectoire, l'augmentation de la vitesse du nœud sur une même trajectoire diminue la quantité de données pouvant être transmise puisque la trajectoire est parcourue dans un délai plus petit. D'autre part, pour une même vitesse, la diminution de $P_{sending}$, sur la même trajectoire, augmente la quantité de données. Ainsi, contrairement aux autres variantes, EXPLoRA-AT, basé sur la quantité de données pour la stratégie d'allocation, donne un PLR important lorsque $P_{sending}$ diminue ou que la vitesse du nœud augmente. Cela explique les valeurs du taux de perte pour EXPLoRA-AT dans la Table III.4 (D_1 et D_2). Pour D_3 et D_4 , le PLR est nul en raison du mode alloué pour l'égalisation de ToA des 5 nœuds, si $P_{sending}$ diminue de plus de 1pkt/17min en utilisant un modèle de mobilité "carré réduit", le PRR de l'EXPLoRA-AT diminue.

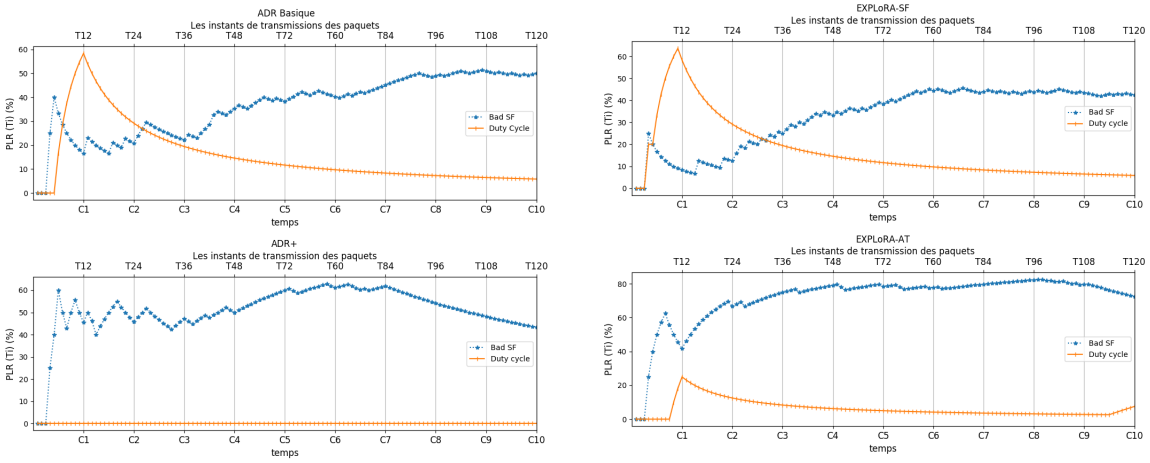


FIGURE III.17 – L'évolution du PLR pour D_1

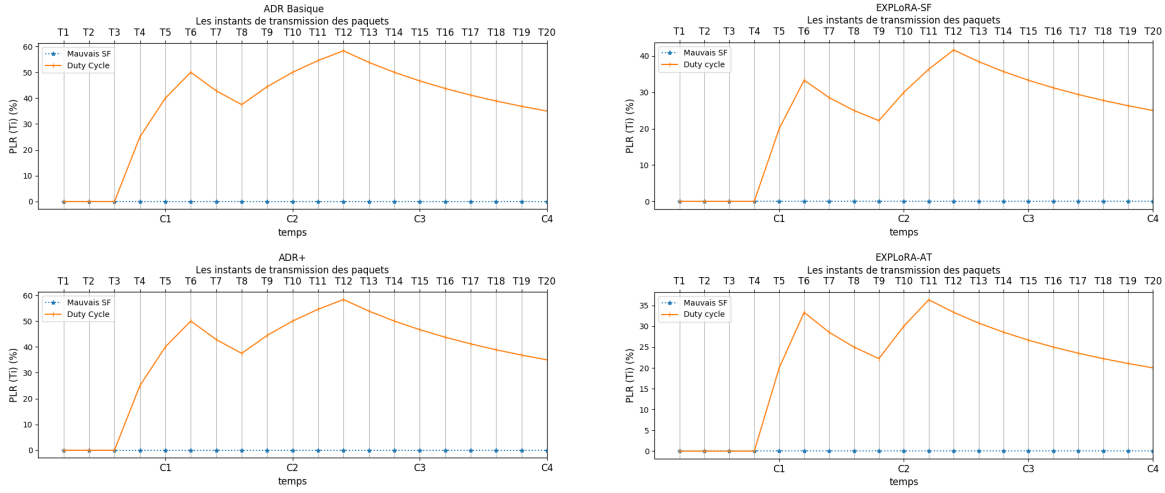


FIGURE III.18 – L'évolution du PLR pour D_5

Pour mieux comprendre quand et comment les paquets sont perdus (Mauvais SF) ou

non-transmis (dépassement de Duty Cycle), nous avons tracé l'évolution du taux de perte de paquets en distinguant entre les deux différentes causes de pertes : échec de retransmission (Mauvais SF) ou dépassement de duty cycle (Duty cycle). Les Figures III.17 et III.18, correspondantes respectivement à D_1 et D_5 présentent l'évolution du taux de perte de paquets dans le temps (PLR(T_i)), où chaque point correspond au $i^{ème}$ paquet envoyé à l'instant (T_i).

Nous avons distingué entre le taux de perte dû au mauvais choix de SF calculé par le nombre de paquets perdus (Mauvais SF) sur le nombre des paquets transmis pendant cette période (courbe en bleu) et le taux de perte dû au dépassement du duty cycle calculé par le nombre de paquets non-transmis (dépassement de Duty Cycle) sur le nombre des paquets transmis (courbe en orange).

Il est à noter que les variantes ADR ne prennent pas en charge la mobilité d'une manière rapide. En effet, lorsqu'un noeud se trouve loin des gateways puis s'en rapproche, les RSSIs faibles se trouvant dans l'historique des RSSIs provoquent l'attribution d'un SF plus élevé (faible débit) que celui dont le noeud a besoin. D'autre part, une fois que le noeud est proche des gateways et commence à s'en éloigner, les RSSIs élevés sont dans l'historique des RSSIs entraînant l'attribution d'un faible SF (sous-estimé) induisant plus de pertes et donc un nombre de retransmissions important.

Dans la Figure III.17, D_1 connaît un nombre très élevé de retransmissions pendant le premier cycle (C_1) en utilisant (ADR Basique, EXPLoRA-SF et EXPLoRA-AT), ce qui se traduit par un PLR élevé en raison du dépassement du Duty Cycle pendant ce cycle. Ceci est indiqué par un pic (courbe en orange). Le PLR élevé est également dû au mauvais SF pendant tous les cycles, sauf pour les trois premiers paquets (courbe en bleu).

Dans la Figure III.18, nous observons que seul le dépassement du Duty Cycle entraîne des pertes de paquets. Cela signifie également que les retransmissions ont permis de compenser les pertes dues au mauvais SF pour D_5. Cependant, en essayant d'empêcher la perte due au mauvais SF par la retransmission, le nœud a dépassé le Duty Cycle durant les cycles C_1 et C_2 (induisant deux pics dans la Figure III.18), ce qui a entraîné un PLR élevé causé par les "paquets non transmis".

Ainsi, bien que les retransmissions donnent la chance à certains paquets d'être retransmis mais causent la perte d'autres puisqu'un nœud dépassant le Duty Cycle entraîne un PLR accru en raison des "paquets non transmis".

3.2.2 Évaluation du ToA et de la consommation énergétique

Le temps de transmission (ToA) ainsi que la consommation énergétique sont aussi des critères importants dans l'évaluation de performance de la QoS dans le protocole LoRaWAN. Nous rappelons qu'un noeud LoRa ne peut pas dépasser 36 sec de transmission par cycle d'une heure.

Dans la Table III.5, les modes d'allocations après les 3 premières transmissions pour chaque noeud sont présentés afin d'avoir une idée sur le principe d'attribution des modes selon les différentes stratégies.

Pour l'ADR Basique, l'allocation est faite selon la plus grande valeur de la liste des SNRs reçus donnant lieu au $Mode10$. Pour ADR+, la moyenne de toutes ces valeurs aboutit à un résultat correspondant au $Mode9$. Concernant EXPLoRA-SF, les 5 noeuds sont classifiés en groupes selon leurs valeurs RSSIs, et D_1 est classifié dans le groupe correspondant au $Mode8$.

Table III.5. La première allocation de mode pour chaque noeuds

Node	ADR Basique	ADR +	EXPLoRA-SF	EXPLoRA-AT	E-ADR
D_1	mode 10	mode 9	mode 8	mode 6	mode 6
D_2	mode 9	mode 8	mode 7	mode 4	mode 5
D_3	mode 6	mode 5	mode 6	mode 3	mode 10
D_4	mode 5	mode 4	mode 5	mode 2	mode 10
D_5	mode 3	mode 2	mode 3	mode 9	mode 8

Utilisant EXPLoRA-AT, les 5 noeuds sont alloués des modes différents afin d'équilibrer les ToAs de tous les noeuds. Pour cela, les noeuds envoyant une grande quantité de données seront alloués un petit SF, tandis que les noeuds envoyant une petite quantité de données seront alloués un grand SF. Dans notre scénario, en classifiant les 5 noeuds selon la taille des données à transmettre, nous remarquons que D_5, transmettant la plus grande quantité sera attribué un petit SF ($Mode9$) (Voir Table II.1) alors que D_4 transmettant une petite quantité de donnée se verra alloué un grand SF ($Mode2$) (Voir Table II.1).

L'attribution du SF pour les 5 noeuds dépend des valeurs RSSIs reçues pendant leurs déplacements. En effet, étant donné que les variantes ADR sont réactives, lorsque les noeuds se retrouvent loin des gateways et commencent à s'en rapprocher, le SF sera attribué selon l'ancien emplacement des noeuds loin des gateways et ne tient pas compte du fait que ces derniers s'en rapprochent, induisant ainsi une attribution d'un grand SF par rapport aux besoins des noeuds, et donc un ToA plus élevé. Inversement, lorsque ces derniers se trouvent proches des gateways et commencent à s'en éloigner, toutes les stratégies à part E-ADR considèrent les anciennes mesures RSSIs pour l'allocation de configuration des noeuds et attribuent des SFs sous-estimés par rapport aux emplacements de ces noeuds provoquant des retransmissions et des pertes dans certains cas.

Les Figures III.19- III.23 présentent le ToA par cycle pour chaque noeud (présentés dans une échelle logarithmique) pour les différentes variantes ADR. Nous remarquons que E-ADR présente dans la plus part des cas un ToA le plus faible tout en respectant la limite du duty cycle. Par ailleurs,r, nous observons dans les Figures III.19 (D_1) et III.23 (D_5) que les transmissions de D_1 et D_5 ont été interrompues à cause du dépassement du Duty Cycle pendant certains cycles (C_1, C_{10} pour D_1 et C_1, C_2 pour D_5) expliquant les pertes notées dans les Figures III.17 et III.18.

Table III.6. Consommation énergétique des noeuds (J)

Node	ADR Basique	ADR+	EXPLoRA-SF	EXPLoRA-AT	E-ADR
D_1	10.61	10.63	10.63	10.56	6.21
D_2	11.7	11.99	11.87	12.01	6.48
D_3	10.69	11.48	10.61	11.48	6.41
D_4	11.9	12.02	11.93	12.2	7.43
D_5	16.34	16.34	16.21	18.29	14.69

En allouant un SF inadéquat, les différentes stratégies sont confrontées à un grand nombre

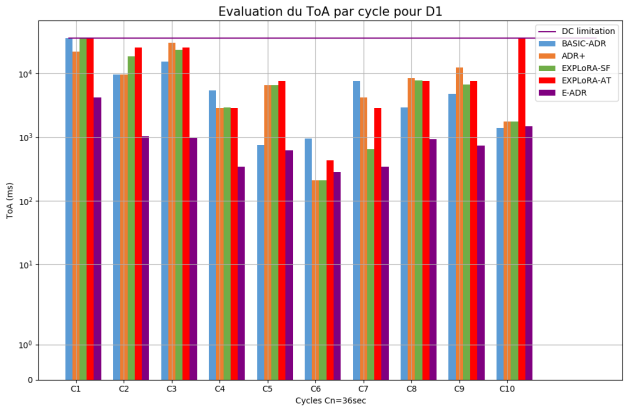


FIGURE III.19 – L'évaluation du ToA pour D_1

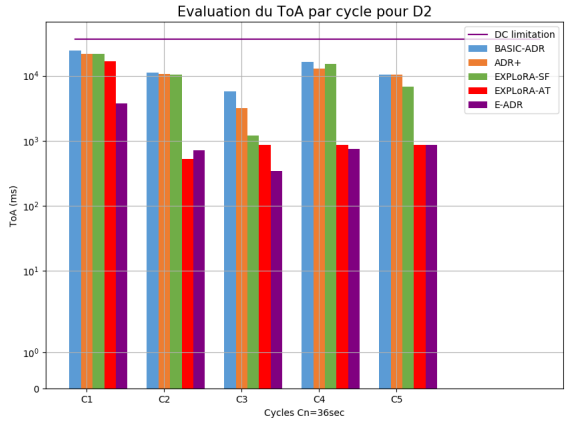


FIGURE III.20 – L'évaluation du ToA pour D_2

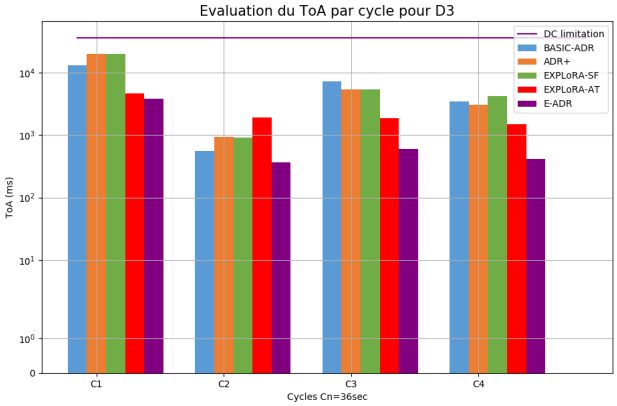


FIGURE III.21 – L'évaluation du ToA pour D_3

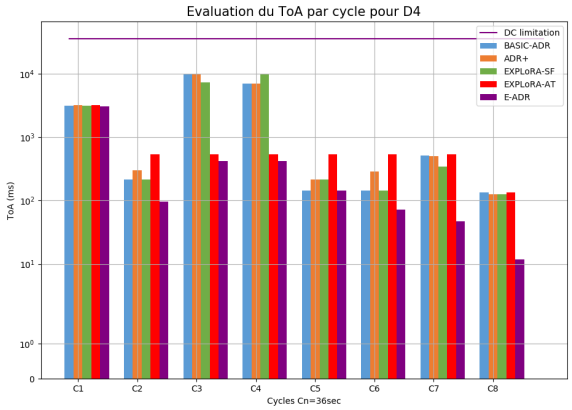


FIGURE III.22 – L'évaluation du ToA pour D_4

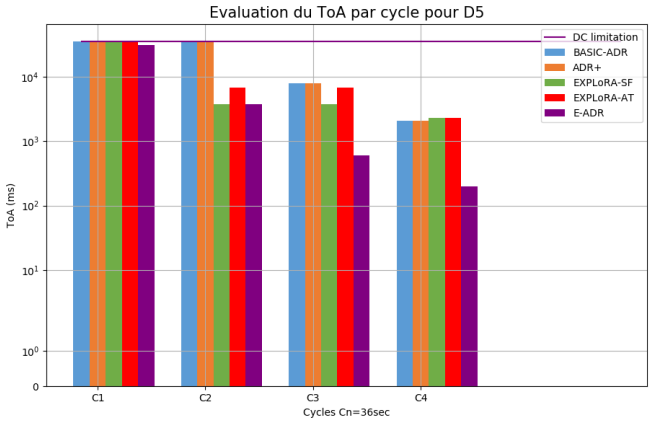


FIGURE III.23 – L'évaluation du ToA pour D_5

de retransmissions et par conséquent à un ToA significatif (Figures III.19- III.23) ainsi qu'une importante consommation énergétique (Table III.6). Ceci explique l'importance de la

consommation d'énergie de toutes les variantes ADR par rapport à E-ADR.

L'E-ADR résulte en un faible ToA par cycle par rapport aux autres variantes ADR grâce à sa stratégie d'allocation proactive basée sur la prédiction des prochaines positions du noeud, ce qui évite les pertes et donc les retransmissions.

3.2.3 Évaluation du délai aller-retour (RTD) moyen

La performance de toutes les variantes ADR en termes de RTD moyen est évaluée et présentée dans la Table III.7. Le calcul du RTD tient compte des différents délais présentés dans la Figure III.1 ainsi que les délais des retransmissions.

De même que pour le ToA, nous constatons que le nombre élevé des retransmissions causé par les mauvais choix des SFs alloués résulte en un RTD moyen très important en utilisant les variantes ADR par rapport à l'E-ADR.

Table III.7. Les mesures du RTD moyen (s)

Node	ADR Basique	ADR+	EXPLoRA-SF	EXPLoRA-AT	E-ADR
D_1	2.91	2.8	2.85 5	3.26	2.28
D_2	3.12	2.97	2.81	2.42	2.1
D_3	2.53	2.52	2.51	2.42	2.09
D_4	2.73	2.75	2.74	2.32	2.14
D_5	11.07	11.07	6.79	7.05	2.81

3.2.4 L'impact des retransmissions

Nous avons pu voir précédemment que les retransmissions peuvent parfois améliorer le PLR dans le cas d'un changement transitif de condition aux dépens d'un RTD plus important. De plus, avec $ADRACKLIMIT = 64$ par défaut, un nombre élevé de retransmissions peut également conduire le noeud à dépasser sa limite de Duty Cycle (en augmentant à son tour le PLR) sans pouvoir augmenter le SF après chaque $ADRACKLIMIT$ retransmissions.

Les Tables III.8 et III.9 montrent l'impact de la variation du $ADRACKLIMIT$ ($ADRACKLIMIT = \{8, 64\}$) sur respectivement la consommation énergétique moyenne par paquet reçu et le PLR pour les différentes variantes ADR.

Table III.8. L'impact du $ADRACKLIMIT$ (m) sur la consommation énergétique moyenne par paquet reçu (J)

Node	ADR Basique		ADR+		EXPLoRA-SF		EXPLoRA-AT		E-ADR	
	$m = 8$	$m = 64$	$m = 8$	$m = 64$	$m = 8$	$m = 64$	$m = 8$	$m = 64$	$m = 8$	$m = 64$
D_1	0.113	0.208	0.11	0.155	0.11	0.155	0.17	0.459	0.051	0.051
D_2	0.26	1.063	0.24	0.5	0.24	0.49	0.17	0.24	0.108	0.108
D_3	0.18	0.23	0.197	0.22	0.167	0.2	0.186	0.201	0.112	0.112
D_4	0.41	0.56	0.426	0.57	0.426	0.56	0.37	0.42	0.256	0.256
D_5	0.845	1.25	0.845	1.25	0.794	1.08	0.771	1.14	0.724	0.724

Table III.9. l'impact du $ADRACKLIMIT$ (m) sur le PLR des noeuds

Node	ADR Basique		ADR+		EXPLoRA-SF		EXPLoRA-AT		E-ADR	
	$m=8$	$m=64$	$m=8$	$m=64$	$m=8$	$m=64$	$m=8$	$m=64$	$m=8$	$m=64$
D_1	32.5%	57.5%	29.16%	43%	30%	47%	56.66%	80.78%	0%	0%
D_2	61.66%	81.67%	31.66%	60%	31.66%	60%	0%	0%	0%	0%
D_3	16.94%	19.28%	5.26%	10.52%	5.26%	8.77%	0%	0%	0%	0%
D_4	13.79%	27.58%	13.79%	27.58%	13.79%	27.58%	0%	0%	0%	0%
D_5	10%	35%	10%	35%	5%	25%	0%	20%	0%	0%

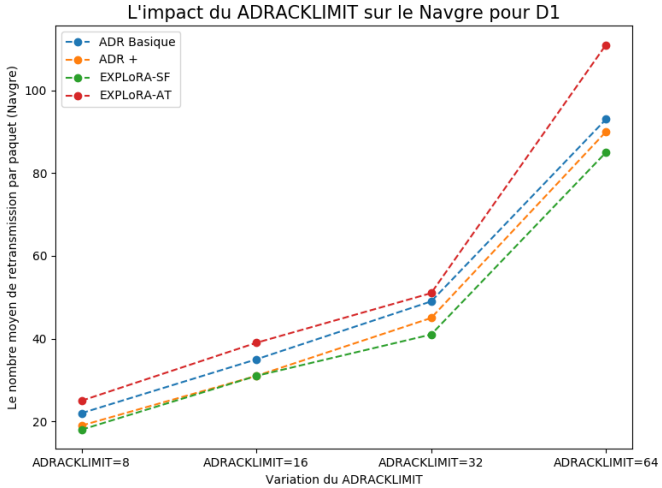


FIGURE III.24 – L'impact de la variation du $ADRACKLIMIT$ sur le nombre de retransmission moyen N_{avgre} pour D_1

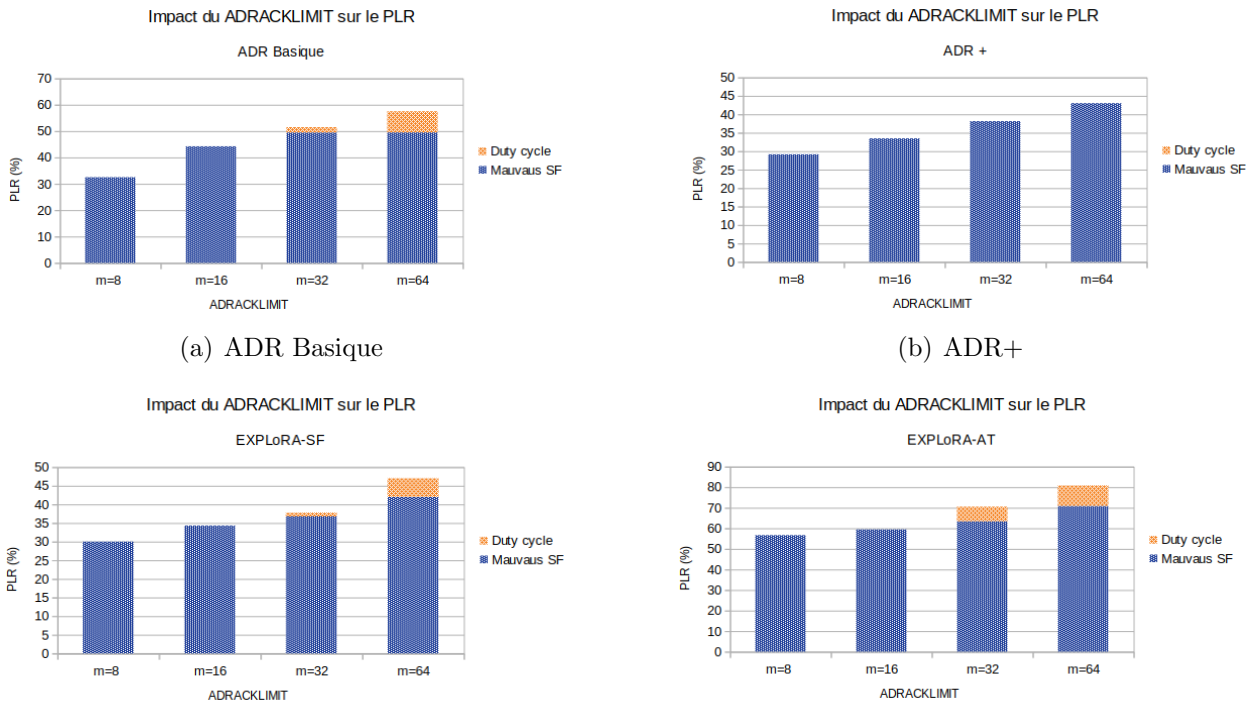
Table III.8 présente la consommation énergétique moyenne par paquet reçu, définie par la consommation énergétique totale sur le nombre de paquets reçus. Comme prévu, le résultat montre qu'en limitant le nombre de retransmissions ($ADRACKLIMIT$ faible) on assure une plus grande efficacité énergétique. Toutefois, les variantes ADR consomment toujours plus d'énergie par rapport à E-ADR, puisqu'elles font toujours recours aux retransmissions.

Table III.9 présente l'impact du $ADRACKLIMIT$ sur le PLR des noeuds. Les résultats montrent que le PLR est réduit en utilisant $ADRACKLIMIT = 8$. Ceci s'explique par le fait qu'en limitant le nombre de retransmissions, on diminue le risque de dépassement du Duty Cycle et donc le taux de perte global.

La Figure III.24 présente l'impact du seuil de retransmission avant l'incrémentation du SF $ADRACKLIMIT$ sur le nombre de retransmissions moyen (dans un intervalle [0 à $\alpha_{i,j}$]). Nous constatons qu'en incrémentant rapidement le SF, le nombre de retransmissions diminue.

La Figure III.25 montre l'impact de la variation du $ADRACKLIMIT$ sur le PLR du noeud D_1 et plus précisément d'observer son impact sur le Duty Cycle en utilisant les variantes ADR Basique, EXPLoRA-SF et EXPLoRA-AT dans ce cas. En effet, D_1 définit plus des pertes à cause du dépassement de Duty Cycle dans le cas d'une faible valeur $ADRACKLIMIT$, par exemple ($ADRACKLIMIT$= 8 ou 16).

Étant donné qu'avec E-ADR aucune perte n'a été notée grâce à un ajustement du SF



(a) ADR Basique

(b) ADR+

(c) EXPLoRA-SF

(d) EXPLoRA-AT

FIGURE III.25 – Impact de $ADRACKLIMIT$ sur le DC pour D_1

en fonction de l'estimation de sa prochaine position, alors E-ADR n'est pas influencé par la valeur $ADRACKLIMIT$.

En résumé, notre solution E-ADR réduit considérablement le PLR, minimise le ToA afin de permettre aux noeuds d'éviter d'atteindre la limitation de 36 sec et au même temps bénéficie d'une meilleure efficacité énergétique par rapport aux autres variantes. Grâce à l'E-ADR et son approche de prédiction des futurs RSSIs, les nœuds n'ont pas eu à retransmettre leurs paquets.

3.3 Cas 2 : Utilisation des messages non confirmés - Pas de retransmissions -

Dans cette partie, nous supposons que les applications ne nécessitent pas de confirmation de la réception, et donc les retransmissions ne sont pas activées. Ainsi, loin de l'impact des retransmissions sur les différents critères de performance, l'utilisation du trafic non-confirmé va nous permettre d'analyser le comportement des différentes stratégies, leurs atouts et leurs limitations. Comme il n'y a pas de retransmissions pour augmenter encore le ToA, nous nous concentrons uniquement sur la comparaison algorithmique des variantes ADR. Ainsi, toutes ces variantes ADR, y compris l'E-ADR, sont testées de manière plus équitable. De plus, ceci nous permettra de voir la différence par rapport au cas 1 et d'analyser l'impact de l'activation et la dés-activation des retransmissions sur la performance du réseau.

Comme dans le cas précédent, la mise à jour de la configuration est exécutée tous les

$n = 3$ paquets à travers un message ACK_{config} programmé en utilisant une fenêtre de liaison descendante chaque $n = 3$ paquets reçus. Ainsi, en ce qui concerne le calcul de la consommation énergétique, une seule liaison descendante est considérée tous les $n = 3$ paquets.

3.3.1 Évaluation du PLR

La Table III.10 présente le PLR enregistré par les différents noeuds utilisant les différentes variantes ADR. Plus de détails, sont présentés dans les Figures III.26 et III.27 qui présentent séparément les pertes dues au mauvais choix de SF et les pertes dues au dépassement du duty cycle pour respectivement D_1 et D_5.

Table III.10. Le PLR des noeuds

Node	ADR Basique	ADR+	EXPLoRA-SF	EXPLoRA-AT	E-ADR
D_1	71.1%	62.5%	60%	92.5%	0%
D_2	86.47%	80%	73.33%	72.47%	0%
D_3	31.57%	28.07%	24.56%	0%	0%
D_4	53.33%	40%	50%	0%	0%
D_5	70%	70%	70%	85%	0%

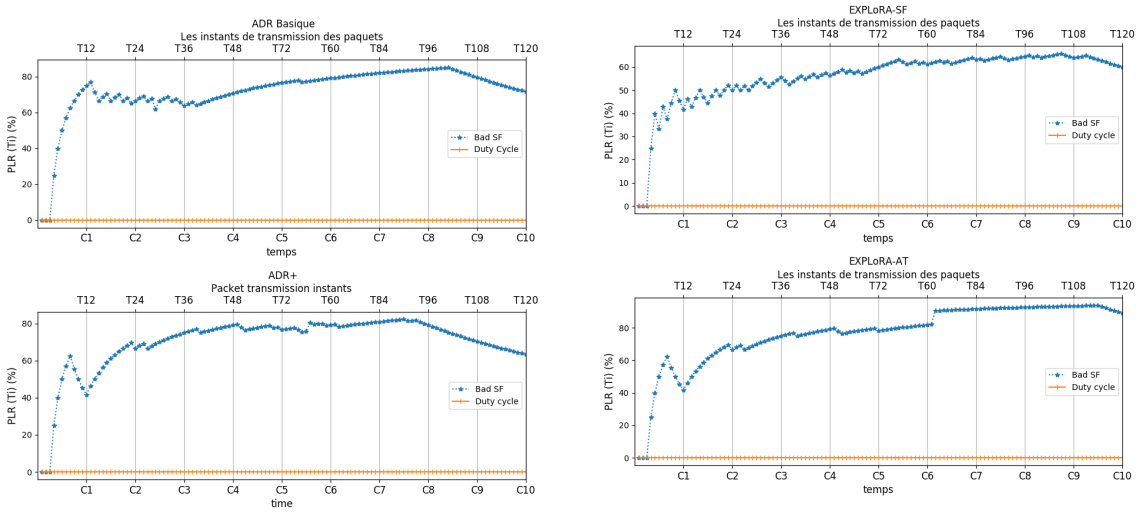


FIGURE III.26 – L'évolution du PLR pour D_1

Nous observons un important PLR pour tous les noeuds dans le cas du ADR Basique, ADR+, et EXPLoRA-SF. Ceci est dû à une mauvaise adaptation à la mobilité et donc à une allocation inadéquate basée sur les anciens RSSIs. Malgré l'allocation de différents SF aux différents noeuds utilisant EXPLoRA-SF, nous remarquons aussi un PLR important à cause de non seulement la mobilité du noeud, mais aussi le fait que les intervalles limite RSSI ne sont pas explicitement considérés.

Pour EXPLoRA-AT, le PLR de D_1 (transmettant 120 paquets) est plus important que celui de D_2 (transmettant 60 paquets). Ceci revient au fait que la stratégie est basée sur

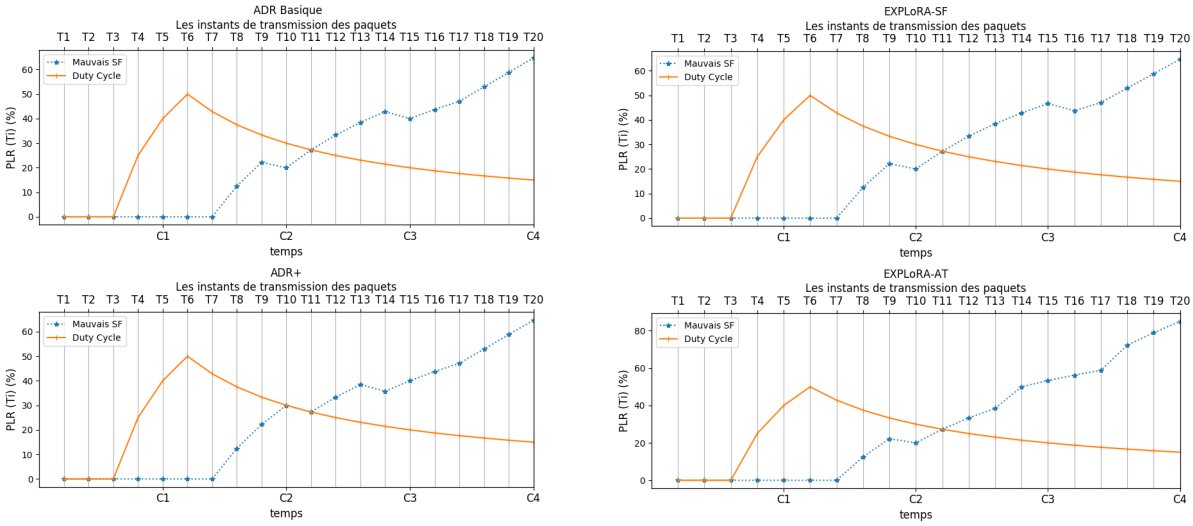


FIGURE III.27 – L'évolution du PLR pour D_5

l'allocation du SF selon la quantité de données afin d'équilibrer le ToA des noeuds et le temps d'occupation des canaux. Cette stratégie résulte en une allocation non-optimale lorsque le noeud envoie une grande quantité de données (avec un grand DR et un petit ToA) entraînant un grand PLR (D_5, par exemple) et d'autre part, la diminution du PLR aux dépens d'une surconsommation énergétique lorsqu'il envoie une petite quantité de données avec un petit DR et un grand ToA (D_3 et D_4, par exemple).

Dans la Figure III.27, nous observons que le grand PLR de D_5 est dû à la limitation du duty cycle (36sec/heure) qui a été dépassé durant le premier cycle (C_1) (à cause de l'éloignement fréquent du noeud par rapport aux gateways et de l'envoi d'une grande quantité de données), et aussi au mauvais SF alloué par les stratégies ADR.

La Table III.10 montre qu'en utilisant E-ADR, tous les noeuds envoient tous leurs paquets grâce à la stratégie d'allocation proactive basée sur la régression linéaire pour la prédiction des futurs RSSIs à partir de la variation des anciens RSSI ($Average_{variation}$).

Comme prévu, Table III.10 montre un grand PLR (à l'exception de l'E-ADR) par rapport à la Table III.4 qui bénéficie des retransmissions pour diminuer le PLR. De plus, comparés aux tests utilisant le trafic confirmé, nous observons que D_1 dans la Figure III.26 ne souffre plus du dépassement de Duty Cycle par rapport à la Figure III.17 (trafic confirmé). De la même manière que D_5, la probabilité de dépassement de Duty Cycle diminue. Figure III.17 montre que D_5 ne dépasse plus le Duty Cycle durant C_2. Cependant, le dépassement de Duty Cycle durant C_1 est lié à l'envoi d'une grande quantité de données et du fait que le noeud s'éloigne fréquemment des gateways. Le PLR causé par le dépassement de Duty Cycle est réduit (de 35% (dans Figure III.17) à 15% (dans Figure III.26) en utilisant l'ADR Basique, par exemple).

3.3.2 Évaluation du ToA et de la consommation énergétique

L'utilisation du trafic non-confirmé mène à une importante diminution du ToA ainsi que la consommation énergétique. Mais ceci ne signifie pas une meilleure performance puisque la

dés-activation des retransmissions mène à une augmentation du PLR comme observé dans la section précédente.

Les Figures III.28- III.32 présentent le ToA cumulatif par cycle (adoptant une échelle logarithmique) pour tous les noeuds utilisant les différentes variantes ADR.

Nous rapportons aussi dans la Table III.11 la consommation énergétique de tous les noeuds..

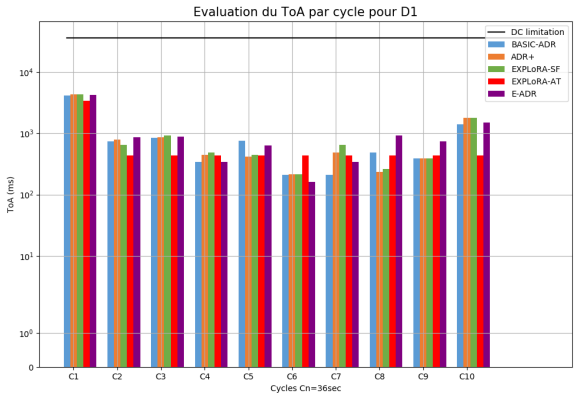


FIGURE III.28 – L'évaluation du ToA pour D_1

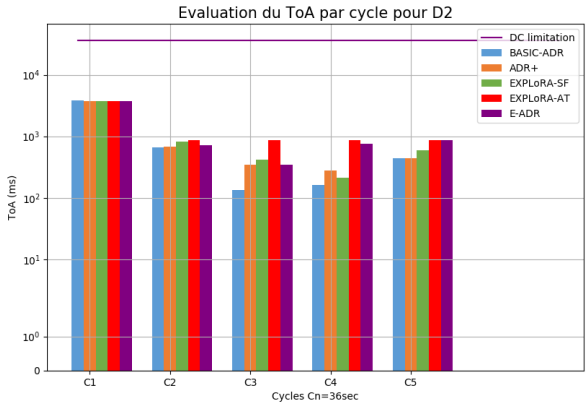


FIGURE III.29 – L'évaluation du ToA pour D_2

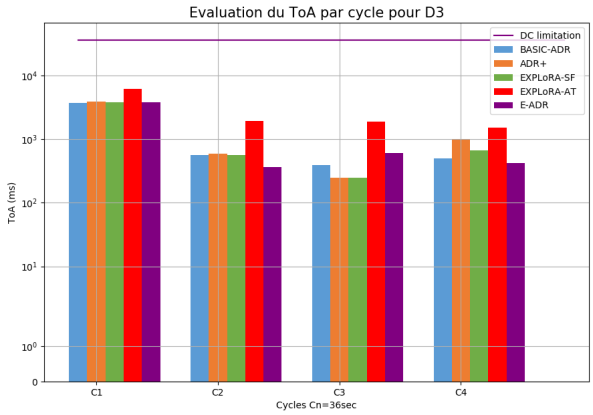


FIGURE III.30 – L'évaluation du ToA pour D_3

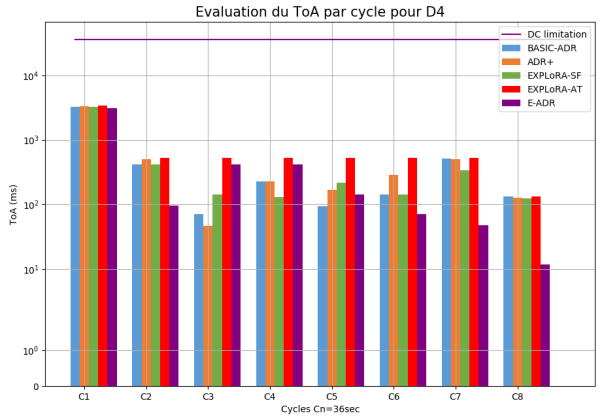


FIGURE III.31 – L'évaluation du ToA pour D_4

Les variantes ADR consomment moins de ToA et moins d'énergie que l'E-ADR dans quelques cas aux dépens d'un PLR important, par exemple pour D_1 et D_2. E-ADR, dans ce cas, incrémente le ToA (ainsi que la consommation énergétique) pour éviter la perte des paquets. Dans d'autres cas, par exemple, pour D_3 et D_4, les autres variantes ADR consomment plus de ToA et d'énergie que l'E-ADR tout en ayant toujours un PLR important à cause d'une sur-allocation du SF.

Le ToA, ainsi que la consommation énergétique (présentée dans la Figure III.11) de D_1 utilisant EXPLoRA-AT, sont plus faibles que les autres variantes, ceci revient au fait que la

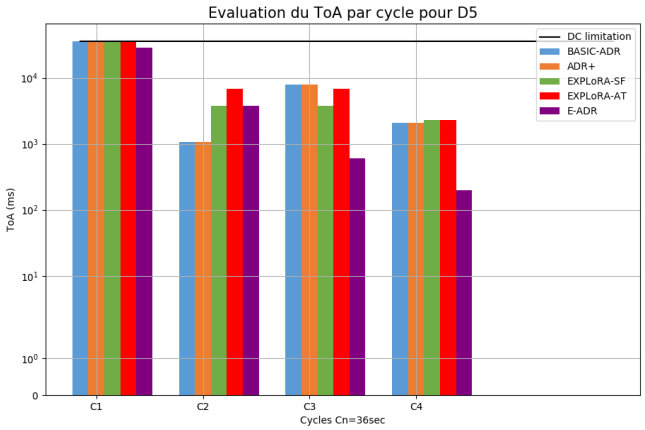


FIGURE III.32 – L'évaluation du ToA pour D_5

Table III.11. Consommation énergétique des noeuds (J)

Node	ADR Basique	ADR+	EXPLoRA-SF	EXPLoRA-AT	E-ADR
D_1	4.96	5.01	5.09	4.67	5.18
D_2	5.06	5.17	5.31	6.34	6.03
D_3	5.13	5.17	5.07	6.13	5
D_4	6.09	6.41	6.28	7.24	5.21
D_5	14.19	14.19	14.15	15.82	14.1

quantité envoyée est importante et d'où l'allocation d'un petit SF (sous-allocation). Pour D_2, même si le ToA est important à cause du SF alloué (petite quantité de données), ceci n'évite pas la perte de paquet, car le SF n'est pas optimal lorsque le noeud s'éloigne des gateways. Pour D_3 et D_4, la sur-allocation du SF, basée sur la faible quantité de données des noeuds, leurs permet de transmettre tous leurs paquets aux dépens d'un grand ToA.

D_5 ayant un important volume de données peut excéder la limite du Duty Cycle (1%) si la configuration n'est pas optimale. Figure III.32 montre qu'en utilisant un modèle de mobilité "carré" et étant très distant des gateways au début des transmissions, l'E-ADR réduit le ToA (et l'énergie) évitant au maximum d'atteindre la restriction du Duty Cycle. De plus, E-ADR permettant à D_5 de finir sa transmission avant d'atteindre les 36 sec mène à une atténuation du PLR (0%). Cependant, en utilisant les autres variantes ADR, D_5 interrompt sa transmission durant le premier cycle C_1 en arrivant à 36 sec induisant un important PLR.

Comparant aux tests utilisant "trafic confirmé", le ToA ainsi que la probabilité d'atteindre la limite ToA, sont réduits (D_1 et D_5) aux dépens d'un PLR important.

Pour résumer, d'une part, même si quelques stratégies bénéficient (points forts) d'un faible PLR quelques fois (EXPLoRA-AT, par exemple), elles occupent les canaux pendant longtemps incrémentant ainsi le ToA (limites) comparant au E-ADR, qui non seulement réduit le PLR mais aussi minimise le ToA. D'autre part, les stratégies bénéficiant d'une grande efficacité énergétique (points forts) souffrent d'un PLR important (limites) résultant en une dégradation de la QoS du réseau. De plus, nous soulignons que toutes les variantes ADR, spécialement l'ADR Basique considérant seulement la valeur allouant un faible SF (RSSI

max), peut faire face à un blocage si le noeud ne se rapprochera pas des gateways tout le long de la trajectoire. Dans ce cas, le SF ne sera pas incrémenté par le noeud puisque sans retransmissions, le noeud ne peut être conscient de la perte de paquets. Dans nos tests, la seule chance pour atteindre les gateways est lorsque le noeud s'en rapproche et que le SF devient suffisant. Finalement, à travers nos tests, nous montrons que l'E-ADR fournit un bon compromis entre la consommation énergétique et la fiabilité de transmissions. En effet, en privilégiant le taux de réussite des paquets, l'E-ADR peut éviter les retransmissions, ce qui permet d'économiser le ToA et l'énergie.

3.3.3 Évaluation du délai de transmission

La performance du E-ADR en termes de délai moyen de transmission est évaluée et comparée aux variantes ADR. Le délai de transmission est le temps mesuré pour transmettre un paquet du nœud au serveur. Sachant que, dans le cas de paquets non confirmés, le délai de transmission ne tient pas compte de la phase d'accusé de réception et que seuls les paquets reçus sont pris en compte pour l'évaluation du délai de transmission. La Table III.12 présente le délai moyen de transmission des paquets des 5 nœuds.

Table III.12. Délai de transmission moyen mesuré pour les 5 noeuds (s)

Node	ADR Basique	ADR+	EXPLoRA-SF	EXPLoRA-AT	E-ADR
D_1	0.57	0.58	0.58	0.56	0.58
D_2	0.58	0.61	0.63	0.71	0.68
D_3	0.6	0.59	0.59	0.7	0.59
D_4	0.7	0.75	0.72	0.74	0.54
D_5	2.89	2.99	2.82	3.24	2.11

Selon la Table III.12, D_1, D_2, D_3 et D_4 connaissent un faible délai moyen de transmission pour toutes les variantes d'ADR à la différence de D_5 qui présente des délais beaucoup plus importants à cause de la taille plus importante des paquets (250 octets au lieu de 25 octets).

Cependant, E-ADR donne les valeurs les plus faibles grâce à sa stratégie d'allocation qui lui permet d'avoir une longueur d'avance sur les autres. Par exemple, lorsque le nœud est loin de la gateway et se rapproche progressivement, le SF est diminué plus rapidement pour E-ADR que pour les autres variantes.

En ce qui concerne la stratégie EXPLoRA-AT, les délais de transmission moyens diminuent avec l'augmentation de la quantité de données transmises au détriment d'un taux de perte important, comme le montre la Table III.12.

Pour synthétiser, selon les tests effectués dans les deux cas (avec ou sans retransmission), bien que l'ADR+, l'EXPLoRA-SF et l'EXPLoRA-AT surpassent l'ADR Basique, l'E-ADR les surpasse en termes de PLR, ToA, efficacité énergétique tout en minimisant la probabilité d'atteindre la limitation du Duty Cycle. Il est donc plus efficace pour s'adapter aux conditions dynamiques.

Toutefois, bien que L'E-ADR présente de meilleurs résultats en termes de QoS, cette évaluation s'est limitée à des modèles de mobilité connus. Compte tenue des environnements intelligents où des applications mobiles à trajectoire aléatoire peuvent être présentes,

telles que les capteurs pour suivi des animaux dans une ferme connectée. Nous proposons dans la section suivante une extension du E-ADR dans le cas de mobilité à trajectoire inconnue.

4 – Évaluation de VHMM-based E-ADR : le cas de noeuds mobiles à trajectoires inconnues

Comme présenté dans le chapitre précédent, nous avons proposé une extension de E-ADR qui pourrait être utilisée avec des mobiles suivant des trajectoires quelconques et inconnues. Il s'agit de « VHMM-based E-ADR » qui sera évalué en termes de PLR, de ToA et de consommation énergétique, ainsi qu'en termes de délai de transmission.

4.1 Scénario de test

Considérons un réseau LoRaWAN composé de 5 noeuds mobiles parcourant une zone S=($[X_{min}, X_{max}] = [-120, 140]$, $[Y_{min}, Y_{max}] = [0, 140]$) utilisant des trajectoires irrégulières et dédiés au suivi de 5 animaux, envoyant 25 octets chaque $P_{sending}(j) = 2$ min et se déplaçant à une vitesse moyenne de 2m/min. Nous supposons que le suivi concerne l'état de santé des animaux malades, nécessitant donc la transmission de messages confirmés.

Les noeuds commencent leurs premières transmissions en utilisant SF12. Nous rappelons que les noeuds utilisent les 10 configurations prédéfinies par Waspmote où la puissance TP est fixée à 14dBm et le taux de codage CR à 4/5 (Table II.1). Les noeuds transmettent leurs données pendant un cycle d'une heure ($C = 1heure$).

Pour une comparaison équitable des différentes stratégies ADR sans l'influence de la fréquence d'exécution de l'algorithme, nous proposons d'exécuter la mise à jour de la configuration après chaque réception de paquet ($n = 1$ paquet). Pour le cas de VHMM-(2,1), les deux premiers paquets seront envoyés en utilisant le SF12 (mode 1) afin de garantir leur réception par les différentes gateways et de pouvoir appliquer le VHMM-(2,1). Ensuite, le processus d'allocation de configuration sera mis à jour après chaque $n = 1$ paquet reçu.

Nous avons construit une phase d'apprentissage en prélevant les différentes positions/Point d'intérêt (POIs) des noeuds durant leurs déplacements. Nous avons enregistré un ensemble de 90 trajectoires pour chaque noeud, qui vont être considérés dans la phase d'apprentissage. Dans les séquences d'observation du modèle VHMM-(2,1), ces différentes trajectoires sont divisées en des groupes contenant chacun 3 POIs pour un HMM-2 ($POI_{(i)}$ →$\text{POI}_{(j)}$ →$\text{POI}_{(k)}$) et des groupes contenant 2 POIs pour un HMM-1 ($POI_{(j)}$ →$\text{POI}_{(k)}$).

Un second ensemble de 5 trajectoires pour chaque noeud est dédié à la phase test. Nous allons durant ces tests prédire le prochain POI de chaque noeud mobile selon les deux POI ancien et courant (HMM-2) ou selon les deux derniers POIs (ancien et courant pour HMM-2) ou selon le POI courant seulement (HMM-1) dans le cas où les deux enchaînements de HMM-2 n'existent pas dans la séquence d'observation (phase apprentissage), ceci dans le but d'éviter une prédiction aléatoire du prochain POI.

4.2 Évaluation de performance

Comme mentionné précédemment, le paramètre $\alpha_{i,j}$ (Eq III.1) définit le nombre maximal de retransmissions possibles du paquet i envoyé par D_j. Ces retransmissions sont autorisées seulement dans un intervalle $[T_i,T_{i+1}]$, où $(i+1)$ correspond à la prochaine transmission, ceci afin de ne pas avoir un décalage d'envoi des paquets programmés (Voir section 1).

4.2.1 Évaluation du PLR

Dans cette partie, nous évaluons les différentes variantes ADR en termes de PLR.

Nous commençons d'abord par présenter dans la Table III.13 le résultat de la première allocation de configuration et à la deuxième transmission pour les autres variante) après réception des valeurs RSSI de chaque noeud selon les quelles le serveur alloue la prochaine configuration en utilisant les différentes stratégies ADR (ADR Basique, ADR+, EXPLoRA-SF et EXPLoRA-AT). Cette première allocation concernera la troisième transmission dans le cas de VHMM-(2,1) (ayant besoin de deux POIs précédents) et la deuxième transmission dans le cas des autres variantes.

Table III.13. Allocation de mode pour chaque noeuds

Node	ADR Basique	ADR +	EXPLoRA-SF	EXPLoRA-AT	VHMM-based E-ADR
D_1	mode 10	mode 10	mode 10	mode 8	mode 8
D_2	mode 2	mode 2	mode 2	mode 6	mode 1
D_3	mode 7	mode 7	mode 3	mode 5	mode 5
D_4	mode 9	mode 9	mode 8	mode 4	mode 8
D_5	mode 6	mode 6	mode 4	mode 7	mode 3

Avant l'allocation présentée dans la Table III.13, nous avons enregistré une valeur RSSI de $-112.62dB$ (pour D_1), $-130.43dB$ (pour D_2), $-121.75dB$ (pour D_3), $-116.27dB$ (pour D_4), et $-124.83dB$ (pour D_5). À partir de ces valeurs RSSI, le serveur alloue le mode de configuration à utiliser lors de la prochaine transmission. L'ADR Basique et l'ADR+ se basent respectivement sur la valeur maximale et la valeur moyenne de la liste des SNRs reçus correspondante à la valeur maximale RSSI, mais étant donné que cette liste ne contiendra qu'une seule valeur ($n = 1$), le serveur allouera le même mode à chaque fois pour ces deux stratégies et ça sera $Mode10$ pour D_1 lors de la première allocation, par exemple (Table III.13). Concernant EXPLoRA-SF, les 5 noeuds seront classifiés en groupes selon leurs valeurs RSSI, et dans notre cas D_1 est classifié dans le groupe correspondant au $Mode9$. Utilisant EXPLoRA-AT, vu que D_1 envoie la plus grande donnée, ce dernier sera attribué le $Mode8$. D_5 classé deuxième, en termes de plus grandes quantités de données, sera attribué le $Mode7$. Les 3 autres noeuds (D_2, D_3 et D_4) envoient la même quantité de données, et dans ce cas là, le serveur leur attribue aléatoirement les 3 prochains modes ($Mode6$ $Mode5$ et $Mode4$).

Selon les allocations présentées dans la Table III.13, pour D_1 par exemple, toute configuration $<$ au $mode8$ (VHMM-based E-ADR) signifie une perte de la transmission et toute configuration $\geq$ au $mode8$ signifie une configuration adéquate.

La Table III.14 présente les PLRs moyens obtenus à partir des 5 tests effectués pour chaque noeud D_j utilisant les différentes variantes ADR. Le PLR d'un noeud D_j est défini comme étant le nombre de paquets perdus sur le nombre total des paquets programmés à la transmission. Nous observons que par rapport à la stratégie VHMM-based E-ADR, les différentes variantes résultent en un PLR important même si la mise à jour de la configuration est faite après chaque paquet reçu et que les retransmissions sont activées. Ces retransmissions offrent une chance aux paquets perdus d'être retransmis avec succès, mais premièrement dans certains cas même après $\alpha_{i,j}$ retransmissions, le SF n'est pas suffisant pour atteindre les gateways. Deuxièmement, ces retransmissions entraînent une augmentation du ToA provoquant dans certains cas un dépassement de Duty Cycle. Lorsque le Duty Cycle est dépassé, les paquets restants ne seront pas transmis et seront considérés comme perdus, d'où l'obtention d'un PLR important dans la Table III.14 en utilisant les variantes autres que E-ADR.

Table III.14. PLR moyen sur les 5 tests effectués pour chaque D_j

Node	ADR Basique et ADR+	EXPLoRA-SF	EXPLoRA-AT	VHMM-based E-ADR
D_1	41.66%	36.11%	36.11%	5.55%
D_2	42.85%	47.21%	66.66%	14.28%
D_3	52.38%	57.42%	57.42%	0%
D_4	38.09%	42.85%	9.52%	0%
D_5	42.85%	35.71%	57.14%%	7.14%

Bien que la mise à jour de la configuration et l'adaptation des paramètres est faite rapidement (après chaque paquet reçu), le fait que les stratégies ADR autres que E-ADR s'appuient sur les anciennes valeurs des RSSI des paquets reçus (au lieu de la valeur future tel que VHMM-based E-ADR) entraîne un PLR significatif.

Nous remarquons que l'ADR Basique et l'ADR+ donnent le même taux de perte pour chacun des noeuds en se basant sur une seule valeur SNR (ou RSSI) ($n = 1$) qui est la valeur maximale (en utilisant ADR Basique) et aussi la valeur moyenne (en utilisant ADR+). Dans ce cas, il n'y a aucune différence entre les deux variantes.

EXPLoRA-SF, basée sur l'allocation du SF selon les valeurs RSSI de l'ensemble des noeuds enregistrés dans le réseau, sans tenir compte des limites RSSI, se voit attribué un SF qui, en fonction de la trajectoire du noeud, peut être sous-alloué ou sur-alloué, ce qui explique les PLRs obtenus dans la Table III.14. De même pour la stratégie EXPLoRA-AT, pour laquelle les noeuds envoyant la plus grande quantité se verront attribués un petit SF (plus grand PLR) et les noeuds envoyant une petite quantité auront droit à un grand SF (plus grande consommation énergétique), ce SF peut-être non-adéquat (sous-alloué/sur-alloué) selon l'emplacement du noeud par rapport aux gateways.

Le modèle VHMM-based E-ADR, basé sur VHMM-(2,1), nous permet d'avoir une meilleure précision de la prédiction du prochain POI et donc une allocation de configuration optimale quelque soit la trajectoire.

Pour mieux comprendre les causes de perte paquets (Dépassement de Duty Cycle ou mauvais SF même après $\alpha_{i,j}$ retransmissions), nous avons tracé l'évolution du PLR dans le temps (PLR(T_i)) dans les Figures III.33- III.37. Nous rappelons que comme décrit précédemment, Le PLR(T_i) est défini comme étant : (1) le nombre de paquets perdus

(Mauvais SF) sur le nombre de paquets déjà transmis jusqu'à l'instant T_i (courbe en bleu) et (2) le nombre de paquets non-transmis (dépassement de Duty Cycle) sur le nombre de ceux qui ont déjà été transmis jusqu'à T_i (courbe en orange).

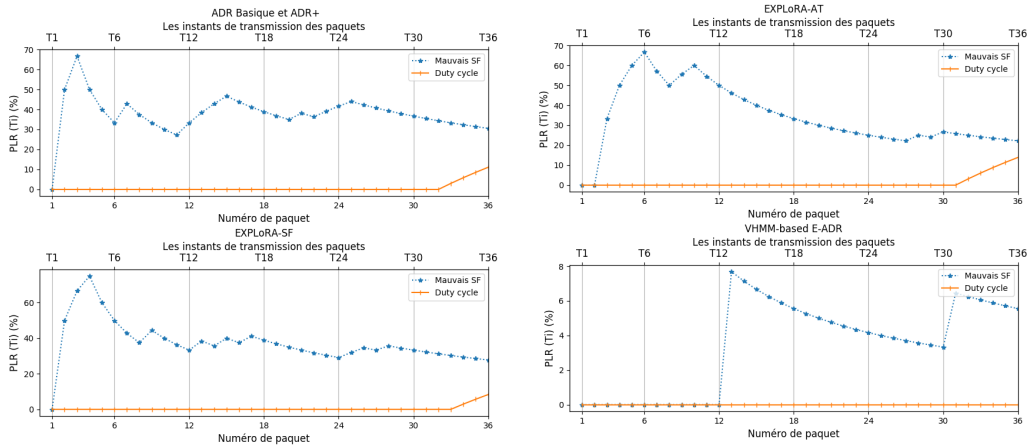


FIGURE III.33 – L'évolution du PLR pour D_1

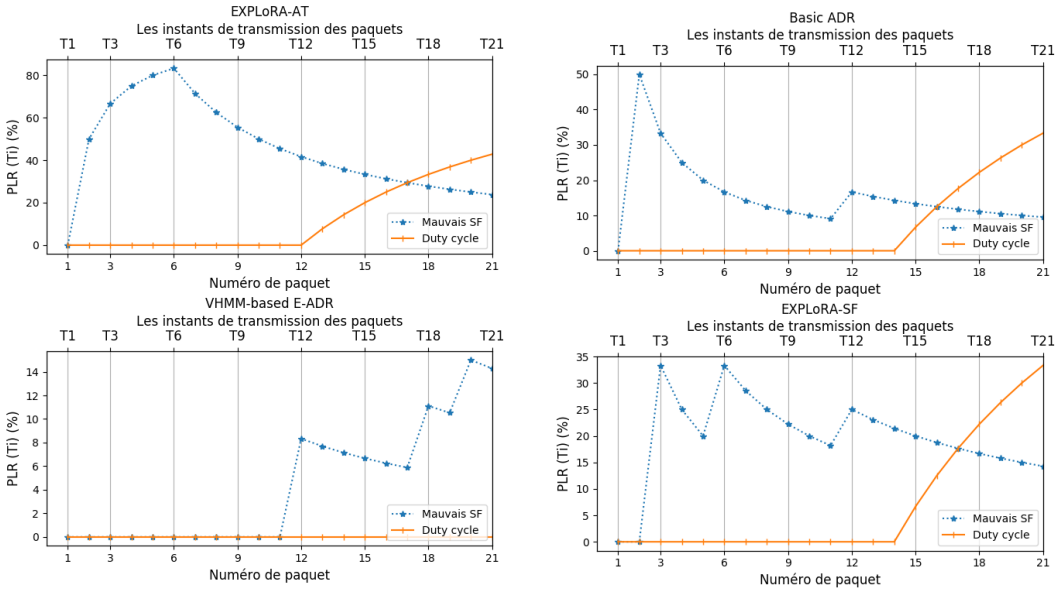


FIGURE III.34 – L'évolution du PLR pour D_2

Les Figures III.33- III.37 montrent que pendant les premières transmissions, les pertes sont principalement dues à la mauvaise configuration allouée au noeuds après $\alpha_{i,j}$ retransmissions et on ne voit des pertes dues au duty cycle qu'une fois que les retransmissions ont considérablement augmentées, engendrant l'épuisement du Duty Cycle. Ces retransmissions sont dues aux stratégies utilisées pour l'allocation de SF qui ne sont pas assez performantes pour s'adapter rapidement aux changements des conditions (mobilité). Cependant, se baser sur une stratégie de prédiction selon le modèle VHMM permet d'avoir moins de retransmissions et donc moins

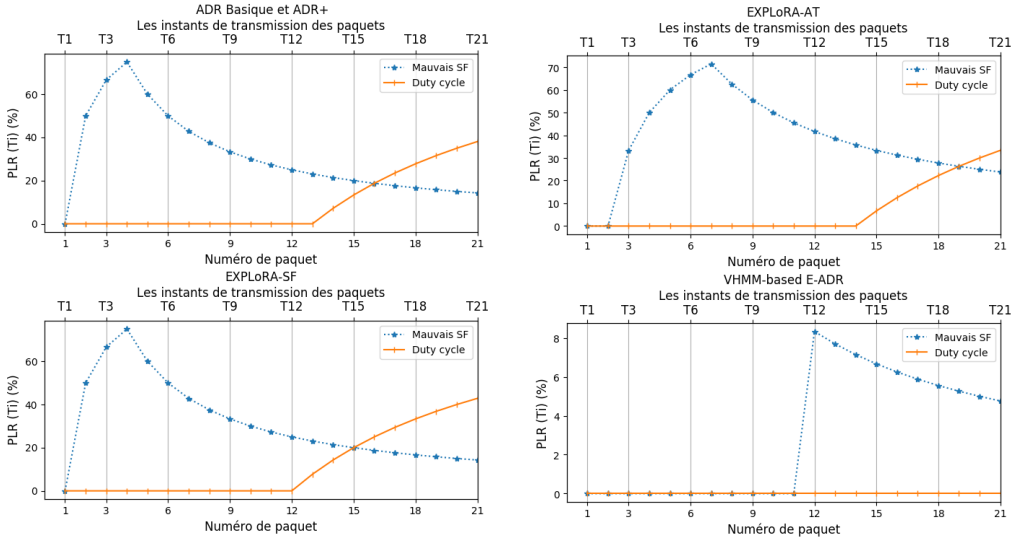


FIGURE III.35 – L'évolution du PLR pour D_3

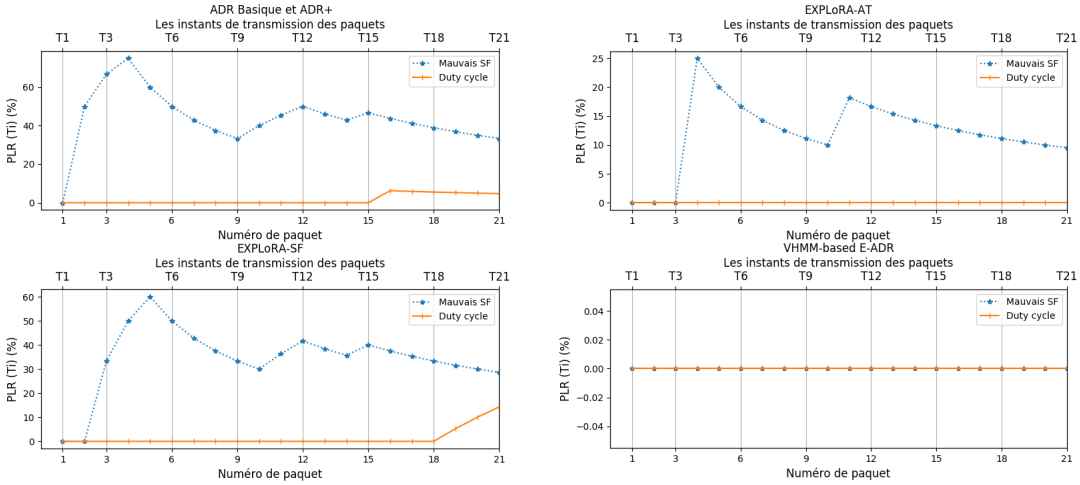


FIGURE III.36 – L'évolution du PLR pour D_4

de pertes et aucun dépassement de duty cycle. Par ailleurs, les pertes notées dans le cas de VHMM-based E-ADR sont dues probablement aux erreurs de prédiction (qui sont assez faibles).

4.2.2 Évaluation du ToA et de la consommation énergétique

Le temps de transmission (ToA) ainsi que la consommation énergétique impactent sur la performance de la QoS du protocole LoRaWAN. Dans cette partie le ToA des noeuds est évalué en utilisant les différentes stratégies ADR.

Les Figures III.38- III.42 présentent le ToA cumulatif consommé par chaque paquet transmis pendant le cycle d'une heure par les 5 noeuds D_j (utilisant une échelle logarithmique) pour les différentes variantes ADR.

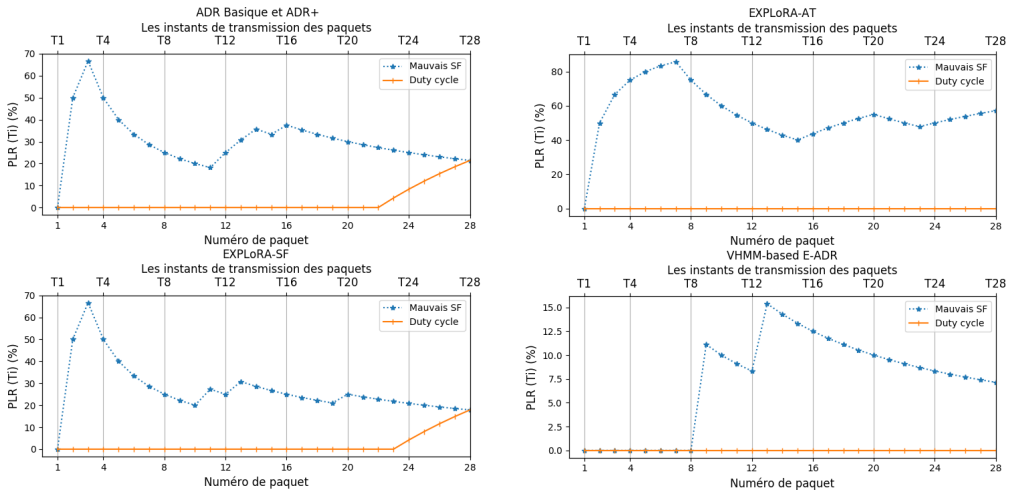


FIGURE III.37 – L'évolution du PLR pour D_5

En attribuant un SF non-adéquat, les noeuds utilisant les différentes stratégies sont confrontés à un grand nombre de retransmissions et donc à une augmentation du ToA ainsi qu'une consommation énergétique significative (Table III.15). En outre, l'augmentation des retransmissions ne signifie pas une atténuation du PLR, car ils sont limités par le duty cycle qu'on ne peut pas dépasser et donc toute variante résultant en un nombre de retransmissions assez élevé, n'ayant pas nécessairement induit une bonne réception de la retransmission, risque d'être confrontée aussi à des pertes dues au dépassement de duty cycle.

Nous pouvons comparer les Figures III.33- III.37 aux Figures III.38- III.42 afin de voir que chaque noeud ayant fait face à un accroissement de la courbe orange (Figures III.33- III.37) se verra dépasser la ligne "Limitation DC" dans les Figures III.38- III.42. De plus, dans la Table III.15 les noeuds ayant dépassé la limite DC auront tous consommé 20.4J d'énergie.

Le VHMM-based E-ADR, n'ayant pas recours à un nombre de retransmissions aussi élevé que celui des variantes ADR, résulte en un faible ToA grâce à sa stratégie d'allocation proactive basée sur la prédiction des prochains POIs à partir d'une phase d'apprentissage, ce qui permet d'éviter au maximum les retransmissions et les pertes au même temps.

Table III.15. Consommation énergétique par paquet reçu (J)

Node	ADR Basique et ADR+	EXPLoRA-SF	EXPLoRA-AT	VHMM-based E-ADR
D_1	0.97	0.88	0.88	0.341
D_2	1.7	1.62	2.91	0.73
D_3	2.04	2.26	2.26	0.26
D_4	1.56	1.7	0.85	0.26
D_5	1.27	1.27	1.17	0.34

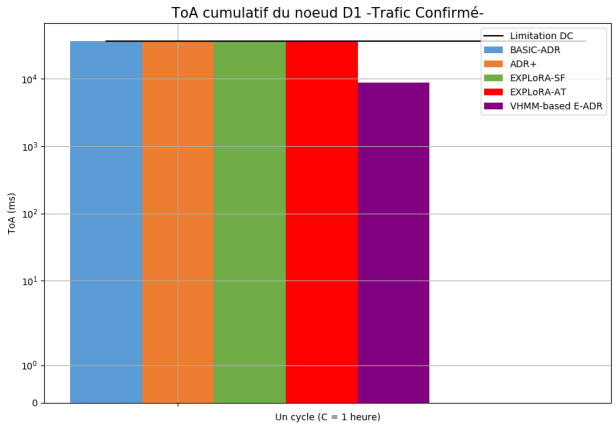


FIGURE III.38 – L'évaluation du ToA pour D_1

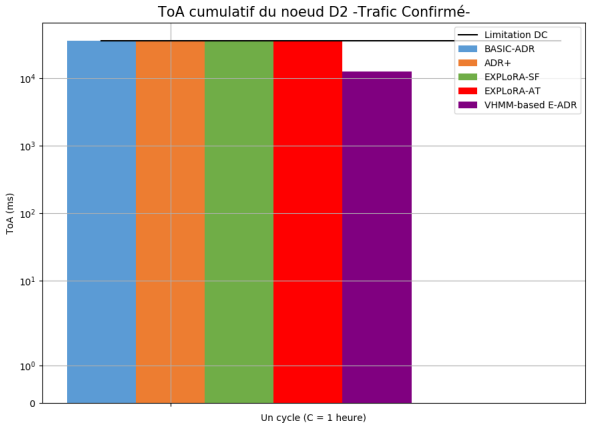


FIGURE III.39 – L'évaluation du ToA pour D_2

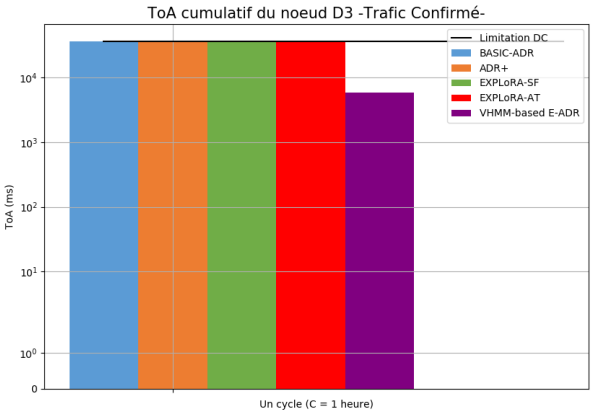


FIGURE III.40 – L'évaluation du ToA pour D_3

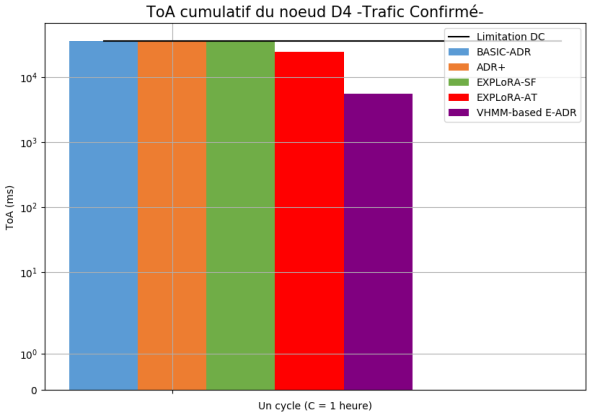


FIGURE III.41 – L'évaluation du ToA pour D_4

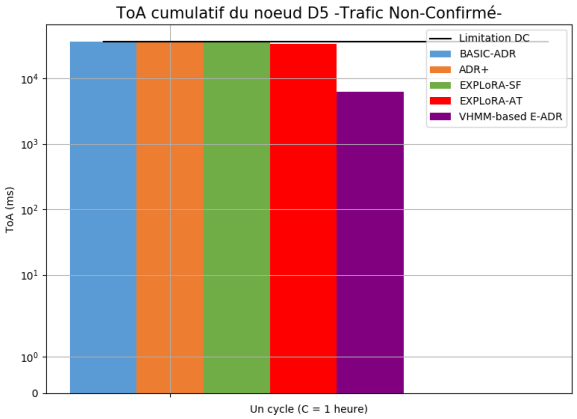


FIGURE III.42 – L'évaluation du ToA pour D_5

4.2.3 Évaluation du délai aller-retour (RTD) moyen

Dans cette partie, le délai moyen aller-retour (RTD) par paquet incluant les retransmissions est évalué pour tous les noeuds utilisant les différentes variantes ADR et présenté dans la

Table III.16. La mesure du RTD dans le cas du trafic confirmé tient en compte les 6 états présentés dans la Figure III.1 ainsi que le délai des retransmissions.

Table III.16. RTD moyen (s)

Node	ADR Basique	EXPLoRA-SF	EXPLoRA-AT	E-ADR
D_1	3.21	3.19	3.26	2.42
D_2	8.56	8.13	5.1	2.59
D_3	6.32	6.03	4.12	2.78
D_4	4.97	4.53	4.17	2.62
D_5	7.84	6.66	3.2	2.81

Nous constatons que les noeuds souffrant d'un nombre élevé de retransmissions causé par la mauvaise allocation faite par les différentes stratégies, résultent en un RTD moyen (Avg) très important. De plus, l'utilisation du VHMM-based E-ADR permet d'optimiser l'allocation de configuration, et donc réduire les retransmissions, ce qui résulte en un faible RTD par rapport aux autres variantes ADR dans la Table III.16.

4.2.4 L'impact des $ADRACKLIMIT$ retransmissions

Comme nous avons pu le noter précédemment, les retransmissions peuvent réduire le PLR au dépit d'une augmentation de la consommation énergétique (par conséquent, suite à une plus grande occupation du canal ToA ou TD). Par conséquent, en utilisant un $ADRACKLIMIT = 64$ par défaut, un grand nombre de retransmissions induisant une grande occupation de canal de transmission, peut provoquer un dépassement de Duty Cycle avant même qu'un noeud puisse atteindre un SF optimal (même après son incrémentation chaque $ADRACKLIMIT$). Les Tables III.17 et III.18 présentent respectivement l'impact du paramètre $ADRACKLIMIT$ en le variant ($ADRACKLIMIT = \{8, 64\}$).

Table III.17. L'impact du $ADRACKLIMIT$ (m) sur la consommation énergétique moyenne par paquet reçu (J)

Node	ADR Basique & ADR+		EXPLoRA-SF		EXPLoRA-AT		E-ADR	
	$m = 8$	$m = 64$	$m = 8$	$m = 64$	$m = 8$	$m = 64$	$m = 8$	$m = 64$
D_1	0.63	0.97	0.61	0.88	0.61	0.88	0.28	0.341
D_2	0.73	1.7	1.11	1.45	1.63	0.58	0.28	0.73
D_3	1.05	2.04	1.2	2.26	1.2	2.26	0.26	0.26
D_4	0.75	1.56	0.91	1.7	0.26	0.85	0.26	0.26
D_5	0.94	1.27	1.19	2.04	0.75	1.52	0.29	0.34

Les résultats obtenus dans les Tables III.17 et III.18 montrent qu'une faible valeur $ADRACKLIMIT$ ($m = 8$) permet d'avoir plus de chance d'atteindre rapidement le SF optimal, car le nœud augmente son SF toutes les retransmissions $ADRACKLIMIT$, évitant ainsi un nombre élevé de retransmissions, un dépassement du Duty Cycle, un PLR important

Table III.18. l'impact du $ADRACKLIMIT$ (m) sur le PLR des noeuds

Node	ADR Basique & ADR+		EXPLoRA-SF		EXPLoRA-AT		E-ADR	
	$m = 8$	$m = 64$	$m = 8$	$m = 64$	$m = 8$	$m = 64$	$m = 8$	$m = 64$
D_1	11.11%	41.66%	8.33%	36.11%	8.33%	36.11%	0%	5.55%
D_2	14.28%	42.85%	19.04%	47.21%	28.57%	66.66%	0%	14.28%
D_3	19.04%	52.38%	28.57%	57.42%	28.57%	57.42%	0%	0%
D_4	4.7%	38.09%	9.52%	42.85%	0%	9.52%	0%	0%
D_5	25%	42.85%	17.85%	35.71%	28.57%	57.14%	0%	7.14%

et une consommation d'énergie élevée. Cependant, même avec un $ADRACKLIMIT$ peu élevé (par exemple, $ADRACKLIMIT = 8$), les variantes ADR consomment toujours plus d'énergie par rapport au VHMM-based E-ADR, puisqu'ils notent plus de pertes et donc plus de retransmissions.

À travers nos tests, nous montrons l'efficacité du VHMM-based E-ADR ayant pour objectif de fournir un bon compromis entre la consommation énergétique et la fiabilité de transmissions, en minimisant les pertes, ainsi éviter les retransmissions et donc augmenter l'efficacité énergétique en réduisant le ToA.

5 – Conclusion

Dans ce chapitre, E-ADR a été comparé expérimentalement aux différentes propositions existantes dans la littérature. Dans un premier temps, nous avons comparé la version basée sur la régression linéaire aux différentes variantes ADR dans un contexte de noeuds statiques en présence d'obstacles mobiles et dans le cas de noeuds mobiles avec des trajectoires connues. Les résultats obtenus confirment que notre approche surpasse les variantes ADR en termes de QoS dans les différents scénarios testés avec et sans retransmissions. D'autre part, l'extension « VHMM-based E-ADR » proposée pour prendre en charge la mobilité quelque soit la trajectoire parcourue a été évaluée. L'extension a été comparée aux différentes variantes ADR en termes de PLR, ToA et consommation énergétique ainsi qu'en termes de délai de transmission. Les résultats ont confirmé l'efficacité du « VHMM-based E-ADR ».

Nos propositions améliorent la performance du réseau LoRaWAN en assurant une meilleure couverture tout en évitant une sur-consommation énergétique grâce à une allocation de configuration optimale. Dans le prochain chapitre, une solution pour réduire le PLR dû au dépassement de Duty Cycle sera proposée ainsi qu'une nouvelle approche d'allocation de ressources sera proposée pour une meilleure amélioration de QoS.

Chapitre IV

Optimisation des techniques d'accès dans LoRaWAN

Dans ce chapitre, nous proposons deux solutions pour l'optimisation des techniques d'accès dans LoRaWAN. La première solution est un mécanisme de Duty Cycle dynamique qui consiste en une meilleure gestion du temps alloué par cycle pour chaque noeud en fonction de son besoin. Le but est de permettre à un plus grand nombre de noeuds de finir leurs transmissions en profitant des ressources non exploitées par d'autres noeuds. La deuxième solution est une approche triplement conjointe « SF-Slot-Canal », permettant d'allouer des ressources d'accès au canal dans LoRaWAN dans le but d'atténuer la probabilité de collisions et de minimiser le taux d'occupation du canal par les éventuelles retransmissions pouvant être causées par les pertes. Ceci permet d'améliorer la QoS globale en termes de taux de perte de paquets et de consommation énergétique. Nous utilisons les modules LoRa Waspmote SX1272 de libelium et STM32 de STMicroelectronics pour l'évaluation de notre proposition dans un contexte de Smart school.

1 – Introduction

Le développement de LoRaWAN a été principalement axé sur les besoins des applications IoT pour répondre à la longue portée et à la longévité des batteries, en négligeant les problèmes de collisions qui peuvent être néfastes pour la qualité de service du réseau. La couche MAC LoRaWAN se base sur la technique d'accès ALOHA pur (P-ALOHA). Cette technique d'accès aléatoire est simple, sa mise en œuvre est rapide et économe en termes d'énergie. En effet, elle ne nécessite aucun mécanisme d'écoute « Listen before talk (LBT) ». P-ALOHA est un protocole asynchrone dans lequel les noeuds émettent leurs données dès qu'elles sont prêtes. Ceci pose généralement le problème de collision entre nœuds. Ce problème est d'autant plus grave quand le nombre de nœuds augmente.

Afin de profiter de la simplicité de ALOHA tout en limitant les interférences, LoRaWAN utilise une couche physique LoRa basée sur l'étalement de spectre (SF) et sur l'utilisation de plusieurs canaux d'émission. Des émissions simultanées avec des SFs différents ne peuvent pas interférer, ce qui réduit le problème d'interférence. Cependant, la sélection des canaux de transmission dans le protocole LoRaWAN est pseudo-aléatoire et le choix initial du SF se fait initialement d'une manière aléatoire ou arbitraire, ce qui n'empêche pas le grand risque de collisions lorsque le nombre de SFs et canaux différents est inférieur au nombre de noeuds qui

transmettent simultanément (cas de réseau très dense).

Par ailleurs, LoRa opère dans la bande de fréquences sans licence ISM, sur laquelle des restrictions sont imposées par les régulations de l'ETSI [21] telles que la durée maximale qu'un équipement peut être actif ou la durée maximale qu'un émetteur puisse émettre par heure. Bien que, l'ETSI permette soit l'utilisation du cycle de service (Duty Cycle) soit la procédure qu'on appelle « Listen Before Talk Adaptive Frequency Agility LBT AFA », la spécification actuelle de LoRaWAN utilise plutôt la technique de Duty Cycle pour être conforme aux régulations en vigueur de la région européenne et les autres régions. Cette restriction impose, pour la bande ISM EU868 MHz, une limite du temps de transmission de 1%, soit 36 sec/heure. Cette limitation entraîne à son tour une augmentation des pertes de paquets. En effet, les retransmissions des paquets perdus à cause des interférences vont causer rapidement le dépassement de la limite du Duty Cycle et donc la perte des données qui ne peuvent pas être envoyées par cycle comme ça a été montré dans les deux chapitres précédents.

Afin de réduire ces pertes, nous présentons dans ce chapitre deux approches : la première consiste à proposer un mécanisme de gestion du Duty Cycle dynamique qui permet une meilleure gestion du temps alloué par cycle pour chaque noeud en fonction de son besoin. Le but est de permettre à un plus grand nombre de noeuds de finir leurs transmissions, donc réduire leurs pertes, en profitant des ressources non exploitées par d'autres noeuds. La deuxième approche consiste à remplacer la technique d'accès ALOHA par un principe d'allocation de slots et de canal de transmission déterministe pour chaque nœud. Ce principe est utilisé conjointement avec nos deux solutions d'adaptation de débit dynamique (E-ADR) et d'optimisation du Duty Cycle (Duty Cycle dynamique) donnant ainsi lieu à une approche triplement conjointe pour mieux gérer les ressources canal (SF-Slot-Canal) .

2 – Approche de Duty Cycle dynamique

Nous avons noté dans les scénarios simulés dans les deux chapitres précédents que dans plusieurs situations, les noeuds se sont confrontés à des pertes dues à l'épuisement de leurs cycles de services ce qui dégrade davantage la QoS de l'application. Par ailleurs, dans les environnements connectés, des applications comme la vidéo-surveillance ou le contrôle d'accès ont besoin d'envoyer une grande quantité de données qui seront malheureusement contraintes par la limitation du Duty Cycle. Cependant, d'autres applications, comme les capteurs d'environnement envoient de très petites quantités d'informations (par exemple, la température, l'humidité, etc.) et sous-utilisent le temps d'activité autorisé de 1%. D'où l'idée de mettre en œuvre un mécanisme de partage du temps d'activité entre les nœuds tel que proposé dans [88]. Ceci permettra aux noeuds qui vont dépasser leurs cycles de service d'emprunter du temps d'activité supplémentaire à partir des nœuds n'allant pas transmettre pendant tous leurs cycles de service. Au lieu d'une allocation de temps de partage FIFO basée sur un temps d'activité global proposé dans [88], qui peut conduire à la privation de nœuds demandeurs en fin de liste ou même certains noeuds de leurs droits d'émettre, nous proposons un nouvel algorithme d'allocation de temps basé sur la classification des différentes demandes en fonction de leurs besoins en termes de QoS. Ceci permet de satisfaire un plus grand nombre de nœuds nécessitant un temps supplémentaire, sans priver d'autres à émettre

leurs données. Notre mécanisme de gestion d'allocation du temps additionnel se base sur différents critères (priorité, niveau de batterie, etc.). Il a été mis en œuvre et testé sur des modules LoRa Waspmote SX1272 de libelium et STM32 de STMicroelectronics, montrant l'amélioration des performances du réseau global.

2.1 Mécanisme de partage du Duty Cycle

Pour introduire plus de flexibilité tout en maintenant la limitation du Duty Cycle total d'une application composée de n capteurs, [88] a d'abord introduit l'idée de permettre à certains nœuds exigeant un débit de dépasser occasionnellement leur limitation de 1% (une sorte de solution de « dernière chance ») tout en maintenant le Duty Cycle global de l'application au-dessous de $n \times 1\%$. Le mécanisme de base, appelé partage du temps d'activité, consiste à diffuser un temps d'activité global (*Global Time*) informant chaque nœud du temps total restant disponible du cycle en cours. Un nœud ayant besoin de plus de temps sera autorisé à utiliser du temps supplémentaire jusqu'à épuisement du *Global Time*.

Ce mécanisme de partage du temps d'activité dans un réseau LoRa à longue portée sans licence tend à faire face au problème de la limitation du temps d'activité dans le cas des applications de vidéo-surveillance. Le mécanisme proposé suppose que tous les noeuds qui participeront au mécanisme de partage s'inscrivent auprès de la gateway LoRa et annoncent leur temps d'activité local (il peut s'agir du temps d'activité total autorisé ou d'une fraction fixe pour tous les noeuds). Ainsi, la gateway calcule le temps d'activité global autorisé à être emprunté qui est la somme des temps autorisés de chaque noeud « *Global Time* » (Eq IV.1) où α est une fraction fixe ($\leq 100\%$) désignant le pourcentage du temps global à autorisé pour l'emprunt, et n le nombre de noeuds dans le réseau. Puis il envoie l'information « *Global Time* » à tous les noeuds qui la partageront. Les noeuds indiqueront la fin de leurs transmissions en utilisant un flag « Flag du dernier paquet », l'activation de ce flag par un noeud permet de savoir que ce dernier n'aura plus besoin d'emprunter un temps supplémentaire du temps global. Cette étape est effectuée à chaque cycle (toutes les heures).

$$GlobalTime = \alpha \times n \times 36sec \qquad 0 \leq \alpha \leq 100\% \tag{IV.1}$$

Tant que ce temps d'activité global le permet, un nœud D_i ayant épuisé son Duty Cycle autorisé (temps d'activité alloué) et ayant besoin de temps supplémentaire pour envoyer ses données peut emprunter le temps restant au temps global. Une vue globale du temps d'activité total restant est maintenue par la gateway LoRa à la réception des paquets et renvoyée aux noeuds dans chaque acquittement (ACK).

Dans [88], l'auteur n'a pas évalué ni proposé de mécanisme pour sélectionner les noeuds qui bénéficieront du temps supplémentaire partagé. En effet, il s'est limité à servir le premier demandeur. En outre, l'auteur suppose que tous les nœuds participant au mécanisme de partage doivent être en attente pour pouvoir recevoir de la gateway les informations actualisées du temps d'activité global et la liste des nœuds concernés par le prêt. Autrement, ils doivent se réveiller périodiquement pour recevoir cette mise à jour. Cela ne correspondrait pas au comportement des nœuds de classe A, mais plutôt à celui des nœuds de classe B, nécessitant une grande consommation énergétique. Cette approche offre en effet une plus

grande souplesse pour mieux gérer la QoS, comme le montre [88]. Elle pose toutefois quelques problèmes supplémentaires. L'un d'eux est le risque de priver les nœuds propriétaires du temps partagé, car la proposition ne définit pas les limites pour éviter la privation de certains noeuds de leur droit d'émettre, en particulier si l'on suppose que les noeuds partagent 100 % de leurs temps d'activité local [88] ou que la gateway LoRa permet d'utiliser le maximum de $GlobalTime$ (c'est-à-dire $\alpha = 100\%$). Les noeuds doivent emprunter un temps supplémentaire à partir du temps libre dans le réseau et ce temps libre pourrait être différent d'un noeud à un autre ce qui fait que mettre à la disposition de tous les noeuds le temps global ou une fraction de temps global n'est pas la solution optimale. Ainsi, un mécanisme d'allocation de temps supplémentaire par une classification prioritaire ou une stratégie qui satisfait un plus grand nombre de noeuds demandeurs en tenant compte de la portée d'un noeud et de son niveau de batterie dans la gestion de l'allocation de temps supplémentaire est meilleure qu'une stratégie FIFO. Un autre problème est lié à la manière de diffuser $GlobalTime$, qui introduit des surcharges et nécessite un mécanisme de réveil radio synchronisé.

Dans la prochaine section, nous décrirons notre solution pour répondre aux enjeux mentionnés ci-dessus.

2.2 Principe du Duty Cycle dynamique

L'idée principale de notre mécanisme est de fournir au noeud qui n'utilise pas tout son temps d'activité maximal autorisés la possibilité de partager son temps de transmission restant avec les noeuds qui doivent dépasser la restriction du Duty Cycle de 1% afin de fournir une meilleure QoS globale. Contrairement à [88], nous nous intéressons aux noeuds de classe A.

Nous supposons que le nombre de paquets que chaque noeud doit envoyer est régulier et est connu au début de la transmission, de sorte que chaque noeud puisse calculer le temps nécessaire par cycle au lieu d'utiliser le « Flag du dernier paquet » comme dans [88]. La gestion de ce temps restant se fera dans le serveur qui n'a pas besoin de diffuser l'information du temps d'activité global restant comme dans [88] puisque le noeud qui décide de partager son temps d'activité restant, ne sera plus concerné par la transmission (il peut même être en veille). Nous permettons également à tous les noeuds de bénéficier de la dynamicité du Duty Cycle lorsque le temps restant est suffisant.

Nous proposons que chaque nœud informe le serveur de son rôle dans le mécanisme de partage pendant le processus d'enregistrement OTAA (Over The Air Activation) [40]. Nous définissons trois rôles :

- Les noeuds « classiques » : ceux qui ne participent pas au mécanisme de partage.
- Les noeuds « donneurs » : ceux qui proposent d'offrir le temps restant de leur Duty Cycle pour qu'il soit consommé par d'autres noeuds.
- Les noeuds « demandeurs » : ce sont les noeuds dont le temps d'activité maximal autorisé (Duty Cycle) n'est pas suffisant pour transmettre leurs données dans un cycle d'une heure. Ils tentent de bénéficier du temps additionnel offert par des donneurs potentiels.

Pendant la procédure de jointure, dans le Join Request, le type de noeud est encodé dans le champs RFU du champs $MHDR$ (Voir Chapitre 1, Section 3.4.6.1, Figure I.7), le temps

à donner ou à demander par le noeud est encodé dans un champs supplémentaire de 6 Bytes rajouté dans le paquet Join Request. Cependant, les noeuds peuvent également préciser ou modifier leurs rôles (demandeur ou donneur) pendant la phase de transmission. Dans ce cas, l'ajout d'un champ supplémentaire de 6 Bytes n'est plus nécessaire, nous ajoutons une nouvelle *commandeMAC* dans le champ $FOpts$ à partir de la plage des CID libres (Voir Chapitre 1, Section 3.4.6.1, Figure I.10) [21]. La commande MAC avec CID=$0x0B$ nommée $DCsharingReq$ sera transmise dans le paquet par le noeud vers le serveur encodant le temps à donner (T_{Don}) ou à demander (T_{Dem}) par le noeud.

2.2.1 Gestion du partage du Duty cycle côté noeud émetteur

Lors de la phase d'enregistrement ou éventuellement en cours de transmission d'un paquet de donnée, chaque nœud souhaitant participer au mécanisme de partage du duty cycle devrait préciser dans le message « Join Request » ou dans le paquet de données envoyé, son rôle en fonction de la taille des données à envoyer pendant ce cycle et donc du temps de transmission (ToA) dont il a besoin. L'estimation du ToA est calculée selon les équations Semtech mentionnées dans la section 3.4.4 du Chapitre 1 (Eq (I.4- I.8)).

La Figure IV.1 illustre le traitement effectué par un noeud pour s'enregistrer auprès du serveur et faire partie du mécanisme de partage.

Si le noeud est un « Donneur » ($ToA_E < ToA_{limit}$, avec ToA_{limit}=temps restant par rapport au duty cycle), il indiquera le temps restant qu'il est disposé à prêter (T_{Don})(Eq IV.2). Dans le cas contraire, s'il s'agit d'un « Demandeur » ($ToA_E > ToA_{limit}$, avec ToA_{limit}=temps restant par rapport au duty cycle), il indique le temps nécessaire à emprunter (T_{Dem}) avant de commencer la transmission (Eq IV.2).

$$T_{Dem}/T_{Don} = |ToA_{limit} - ToA_E| \tag{IV.2}$$

Après la phase d'enregistrement, chaque nœud commence à envoyer ses données selon le protocole LoRaWAN. Le nœud continue à envoyer des informations sur sa participation au mécanisme de partage d'activités tant qu'il n'a pas été satisfait. Par ailleurs, il peut éventuellement modifier son offre ou sa demande de manière dynamique afin de s'adapter à un changement de configuration (ADR ou retransmission à cause d'une perte). Cela permet également à un nœud qui n'a pas participé au mécanisme de partage pendant la phase de jointure, et se rend compte qu'il devrait envoyer plus de messages suite à un évènement, de faire sa demande de temps additionnel pendant la transmission de ses données.

2.2.2 Gestion du partage du Duty cycle côté serveur

Lorsque le serveur reçoit le message d'enregistrement ou un paquet indiquant la participation du noeud au partage de duty cycle (Figure IV.2), il enregistre les informations reçues en fonction du type de chaque noeud dans la table correspondante (table Donneur/table Demandeur). Dans le cas d'un donneur, le serveur met à jour le $T_{G_{Don}}$. Dans les deux cas, un acquittement doit être envoyé au noeud émetteur. Dans le cas d'un demandeur, le serveur vérifie si il existe déjà dans la table (demande non satisfaite au par-avant) afin de mettre à jour seulement le T_{Dem} ou si il s'agit d'un nouveau demandeur afin de lui attribuer une

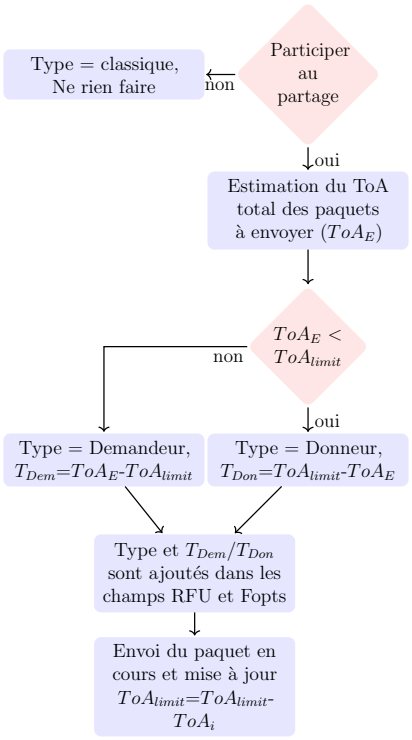


FIGURE IV.1 – Traitement DC-Dynamique au niveau du noeud

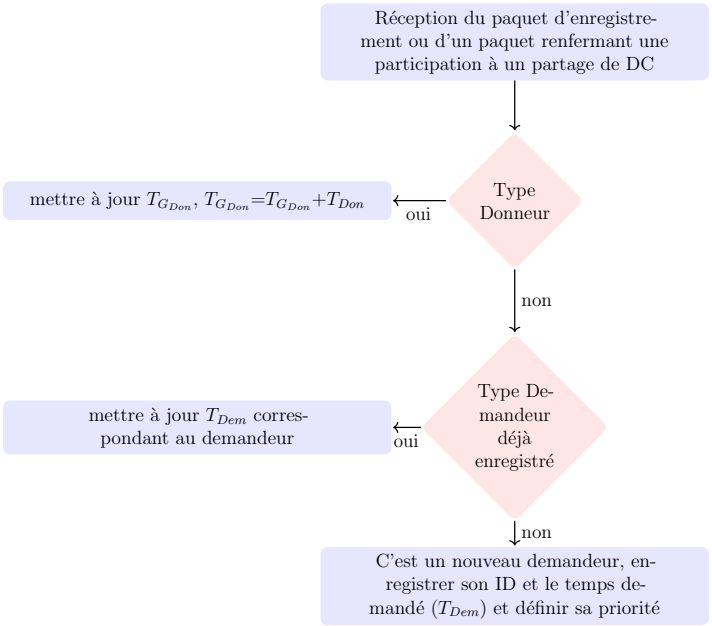


FIGURE IV.2 – Réception d'un paquet avec partage de DC au niveau du serveur

priorité en fonction de certains critères (niveau de batterie, portée, etc.), et vérifie s'il peut satisfaire cette demande.

2.2.2.1 Réponse du serveur à un noeud demandeur

Différents critères peuvent être pris en compte pour classer les noeuds « Demandeur ». Dans notre travail, nous nous intéressons à deux approches :

- La première consiste à servir en premier lieu la demande la plus faible afin de satisfaire le plus grand nombre de demandes.
- La seconde approche consiste à satisfaire d'abord les nœuds dont le niveau de batterie est le plus faible afin d'éviter qu'elles s'épuisent avant la fin de la transmission de ses messages. Dans le cas d'égalité du niveau de la batterie, nous choisissons le nœud le plus proche dont la transmission est la moins lente. La distance de chaque noeud est calculée selon (Eq II.6).

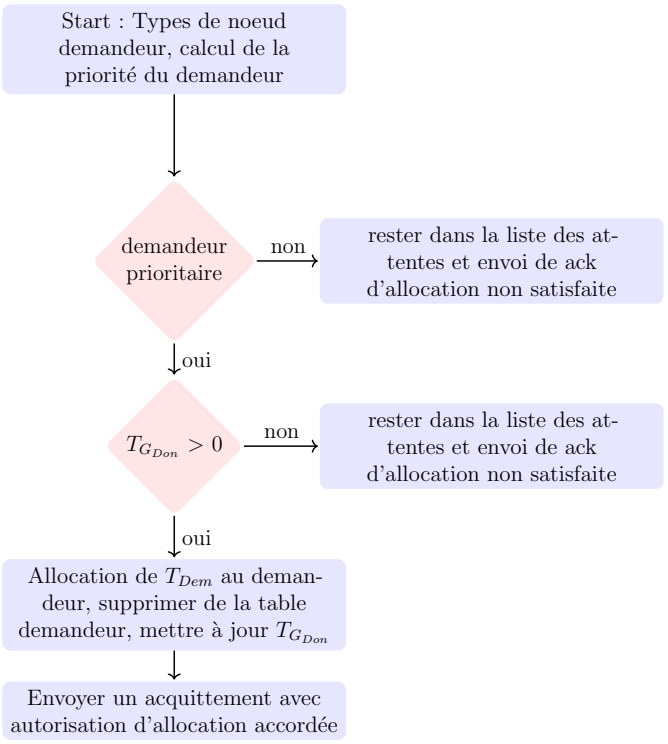


FIGURE IV.3 – Réponse du serveur à une demande de duty cycle supplémentaire

La Figure IV.3 montre le traitement effectué par le serveur après la réception d'une demande d'allocation par un demandeur. Un demandeur envoie un message de type confirmé (avec ACK) lorsqu'il fait une demande de temps supplémentaire afin de recevoir une autorisation d'emprunt à travers cet ACK. Une fois que la demande d'un nœud donné peut être satisfaite (le temps offert par les « Donneurs » est suffisant), le serveur met à jour le temps d'activité restant des « Donneurs » et accorde le temps supplémentaire au « Demandeur » en lui envoyant un message de mise à jour confirmant l'attribution du temps d'activité supplémentaire dans un message ACK. L'acquittement ACK peut préciser que l'emprunt est non satisfait (bit=0), ou l'emprunt est satisfait (bit=1) en l'indiquant dans le champs *RFU* de la trame DL *FCtrl* ((Voir Chapitre 1, Section 3.4.6.1, Figures I.9). Il est à noter que l'ACK doit être envoyé pendant les deux fenêtres de réception (classe A). Si l'attribution

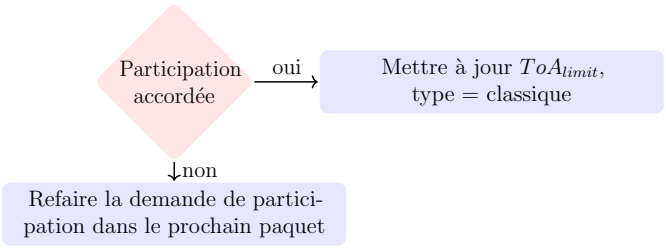


FIGURE IV.4 – La réception d'ACK au niveau du noeud

du temps supplémentaire est accordée, le serveur retire le noeud « Demandeur » de la table correspondante et le noeud demandeur met à jour le temps autorisé d'envoi et son type dans la Figure IV.4. Sinon, le noeud demandeur continue à exprimer sa demande dans sa prochaine trame, si le temps d'envoi autorisé est suffisant pour transmette son prochain paquet.

2.2.2.2 Réponse du serveur au noeud donneur

Lorsque le serveur reçoit un paquet d'un donneur indiquant la participation au partage, il met à jour le $T_{G_{Don}}$ et répond au donneur en lui envoyant un acquittement indiquant l'acceptation de sa participation au mécanisme de partage de duty cycle. Une fois que le noeud donneur reçoit l'acquittement, il met à jour son ToA_{limit} et bascule en un type classique (Figure IV.4).

2.3 Évaluation de la stratégie de partage de Duty Cycle (DC dynamique)

Nous nous sommes intéressés dans nos travaux à différents types d'applications IoT. C'est dans ce contexte que nous proposons d'évaluer cette approche en considérant l'émulation d'un nouvel environnement IoT, correspondant à une école intelligente « Smart school » avec différents types d'applications (surveillance d'une salle de classe, contrôle d'accès à l'école, suivi des enfants dans le bus scolaire, etc.).

Pour évaluer les performances et l'efficacité du mécanisme DC dynamique que nous avons proposé, différents scénarios sont implémentés en utilisant des modules LoRa (noeuds LoRa SX1272 waspmote et STM32 et une gateway LoRa SX1272 waspmote) [49, 83, 84]. Le but est de montrer l'efficacité du DC dynamique par rapport au DC fixe en termes de PLR et du nombre de clients satisfaits du partage de DC. Dans les scénarios qui suivent, l'ADR est désactivés durant tout le cycle.

2.3.1 Scénario avec Duty Cycle fixe (S1)

Nous envisageons un premier scénario simple « Smart School ». Le premier noeud (D_1) est dédié à la surveillance d'une salle d'examen. Une image de 2000 Bytes doit être envoyée en plusieurs paquets en raison de la contrainte de la longueur maximale de la charge utile du paquet LoRa qui varie selon le mode de configuration utilisé. Le second noeud (D_2) est un simple capteur de température dans l'école qui envoie un message de 100 Bytes contenant la température, la date et l'heure.

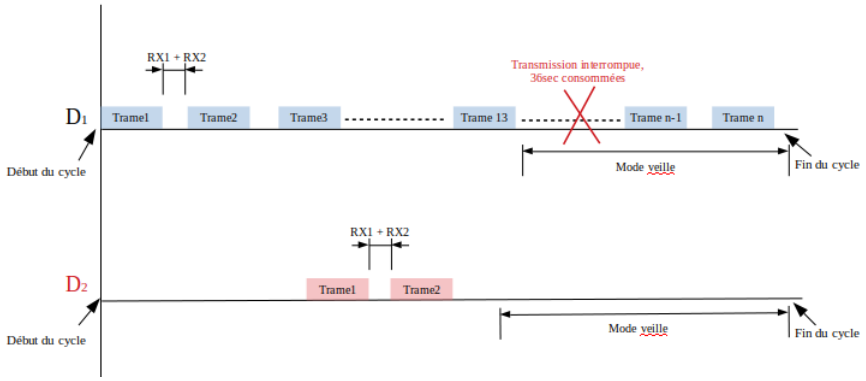


FIGURE IV.5 – Gestion du Duty Cycle dans la transmission d'un noeud

Chaque noeud est limité par un temps de transmission maximal « Limite ToA = 36 sec/cycle ». Les noeuds utilisent la configuration LoRa *mode*1, c'est-à-dire qu'ils utilisent BW= 125kHz et SF= 12 (Table II.1) et une taille maximale de charge utile de 59 Bytes (Voir Chapitre 1, Table I.4). Ce choix est fait, car cette combinaison est la plus adaptée à la longue portée et à la présence d'obstacles, ce qui est le cas des zones urbaines, mais aussi pour garantir que le noeud atteigne les gateways. Dans un premier temps, notre objectif est de se limiter seulement aux pertes causées par l'épuisement du "Duty Cycle" dans un scénario de noeuds fixes. Le scénario est illustré à la Figure IV.5.

Après avoir reçu le message « Join Accept », le cycle commence pour tous les noeuds du réseau. En envoyant le treizième paquet, D_1 se rend compte qu'il a consommé tout son temps d'activité autorisé (36 sec) et qu'il a donc atteint la limite ToA. La transmission des paquets suivants est interrompue et il entre en *mode veille* jusqu'à la fin du cycle. La même chose se produira pour les données qui devraient être envoyées au cours du deuxième cycle. Nous supposons que les paquets non-transmis pendant un cycle d'une heure sont perdus afin d'éviter l'augmentation du délai de transmission. Donc, dans ce scénario de DC fixe, D_1 fait face à un PLR de 38.24% dû au dépassement de DC.

Contrairement au premier noeud, D_2 envoie son paquet et attend un ACK, ensuite il entre en *mode veille* car il a terminé sa transmission au cours du premier cycle sans profiter de tout son temps de transmission autorisé.

2.3.2 Scénario avec Duty Cycle dynamique (S2)

Nous reprenons le même scénario dans le cas d'un DC dynamique. Cette fois-ci, la procédure de jointure est associée à une étape d'enregistrement permettant de manifester le type de noeud dans le réseau (donneur ou demandeur et sa durée de partage) en utilisant le champ RFU dans MHDR (Voir Chapitre 1, Section 3.4.6.1, Figure I.7). Les demandeurs bénéficieront de l'algorithme du DC partagé tant que des donneurs s'enregistrent dans la phase d'enregistrement. Dans la Figure IV.6 D_1 bénéficie d'un acquittement positif (ACK1) signifiant que le serveur l'autorise à consommer un temps supplémentaire emprunté du donneur D_2. D_1 est satisfait et profite d'une atténuation du PLR (0%).

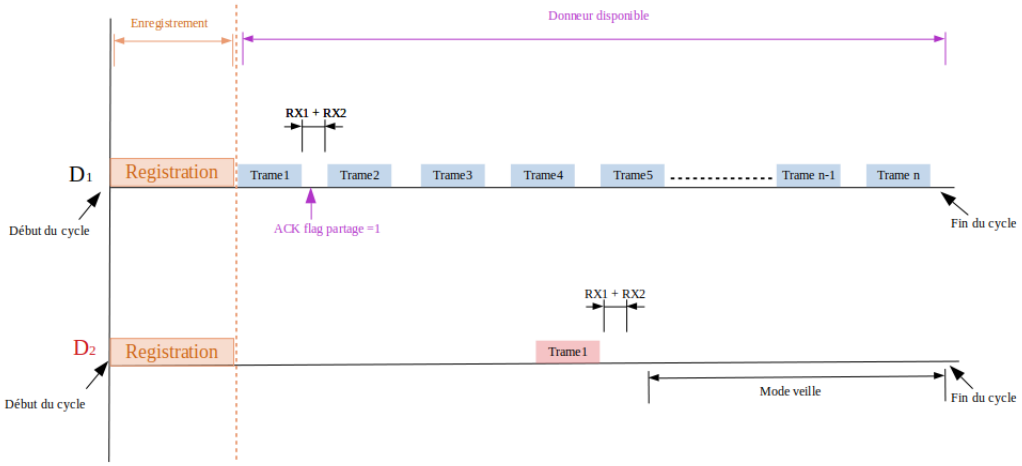


FIGURE IV.6 – Duty Cycle dynamique

2.3.3 Duty Cycle dynamique - gestion de l'allocation

Dans le but de mieux bénéficier du partage de DC dynamiquement, nous intégrons différentes stratégies de gestion du temps supplémentaire à donner aux demandeurs. Ces stratégies reposent sur des priorités qui peuvent être manifestées par le noeud ou attribuées par le serveur lors de l'enregistrement des noeuds. Dans cette section, nous testons deux stratégies de satisfaction des demandes. Dans un premier cas la priorité est basée sur la quantité du trafic à envoyer et dans le second cas sur le niveau de batterie et la portée du noeud.

2.3.3.1 Priorité au faible trafic (S3)

Nous disposons de 5 noeuds (3 demandeurs et 2 donneurs). D_1 est dédié à l'identification des élèves à l'école, il envoie 200 paquets de 7.5 Bytes chacun durant $C = 1heure$. D_2 est dédié à la surveillance (capture d'image) et envoie 8 paquets de 250 Bytes chacun durant $C = 1heure$. D_3 est dédié à l'identification du staff et envoie 50 paquets chacun de 20 Bytes durant $C = 1heure$. D_4 est dédié à la mesure de la température (date, heure et température) et envoi un message de 100 Bytes durant $C = 1heure$. Enfin, D_5 prévient l'administration du nombre de courriers dans la boite aux lettres et envoie 10 paquets de 70 Bytes chacun durant $C = 1heure$.

Durant nos tests expérimentaux les noeuds émettent leurs paquets en utilisant le $mode1$ à des instants différents afin d'éviter les interférences et ne considérer que les pertes dues à la limite du DC.

Les trois demandeurs ont besoin respectivement de 25.56 sec, 46.08 sec et 5.04 sec. Les donneurs D_4 et D_5 partagent respectivement : $31,89$ sec et 7.27 sec. Ce temps additionnel (des donneurs) est partagé sur les noeuds demandeurs selon une stratégie d'allocation, offrant « la priorité la plus élevée au plus faible trafic ».

Le serveur commence par satisfaire D_3 puis met à jour le temps donné : 34.12 sec. Ensuite, il répond à D_1 et met à jour le temps donné : 8.56 sec. Mais en arrivant à D_2, le serveur réalise que le temps donné ne peut pas satisfaire la demande de D_2. Cette stratégie d'allocation a

permis la satisfaction d'un plus grand nombre de noeuds. Si la priorité était au trafic dense, le nombre de noeuds satisfaits serait plus faible.

2.3.3.2 Priorité au plus faible niveau d'énergie et au noeud le moins distant (S4)

Le même scénario est testé, mais selon une autre stratégie de priorité. La priorité revient aux noeuds ayant le plus faible niveau de batterie. D_1, D_2 et D_3 envoient leurs paquets d'enregistrement en tant que demandeurs (3 demandeurs). En recevant les demandes des noeuds D_1, D_2 et D_3 pendant la phase d'enregistrement, le serveur fixe une priorité pour chacun d'eux en fonction d'un certain critère (portée, niveau de batterie, application urgente, etc.). Dans notre test, nous avons considéré comme exemple le niveau des batteries qui peut être communiqué à travers la commande MAC *DevStatusAns* [21] et les distances à travers les mesures RSSI. Ainsi, la priorité la plus élevée est accordée au noeud ayant le niveau de batterie le plus bas et, en cas d'égalité du niveau de batterie, la priorité serait au noeud le plus proche.

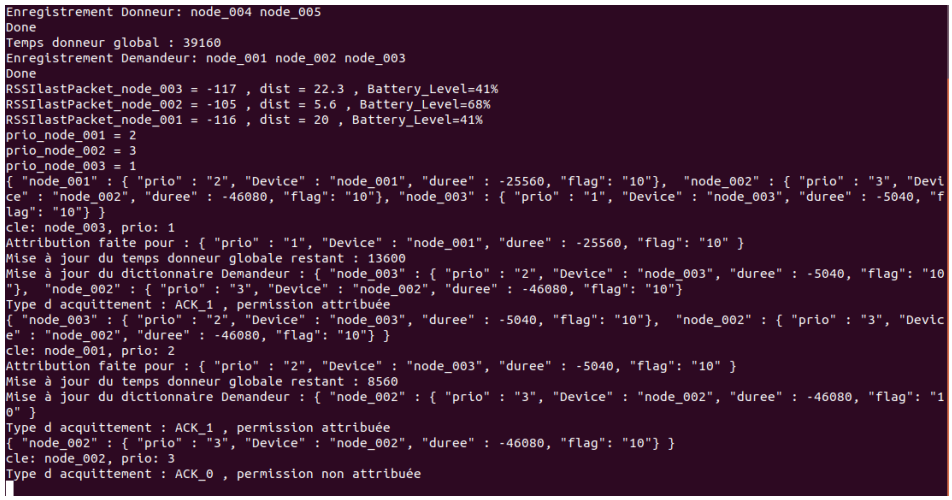


FIGURE IV.7 – Résultat du scénario S4 sur le terminal du serveur

La Figure IV.7 présente le traitement du serveur pour le scénario S4. Les noeuds commencent la transmission de leurs premiers paquets, le serveur répondra par un ACK1 au demandeur ayant la plus haute priorité puis passera au demandeur dont la priorité est moindre. Dans notre expérimentation, les noeuds D_1 et D_3 possèdent le même faible niveau de batterie. Donc, le serveur les classifiera en fonction de leurs distances par rapport à la gateway et D_1 est servi avant D_3.

Après attribution du temps supplémentaire à D_1 et D_3, le temps restant donné est mis à jour par le serveur, ce dernier se rend compte qu'il n'est pas suffisant par rapport à la demande du noeud D_2. D_2 continue l'envoi de ses paquets, mais sa transmission sera interrompue lorsque la limite ToA sera atteinte, il ne bénéficiera pas du mécanisme de DC partagé.

2.3.4 Comparaison DC dynamique / fixe

Dans cette partie, nous évaluons le nombre de noeuds satisfaits dans les différents scénarios en comparant l'utilisation du DC dynamique à l'utilisation de base de 1% de DC. Nous évaluons ensuite le ToA consommé lors de l'ajout des nouveaux champs pour l'enregistrement des types de noeuds. Enfin, nous évaluons le PLR par noeud pour le scénario de gestion de l'allocation (S3).

Les Figures IV.8, IV.9 présentent respectivement : le nombre de noeuds satisfaits dans les scénarios S1 et S2, S4 et S4//. Le scénario S4// reprend le scénario S4 sans utiliser la gestion d'allocation par priorité mais basé sur FIFO (première demande servie en premier).

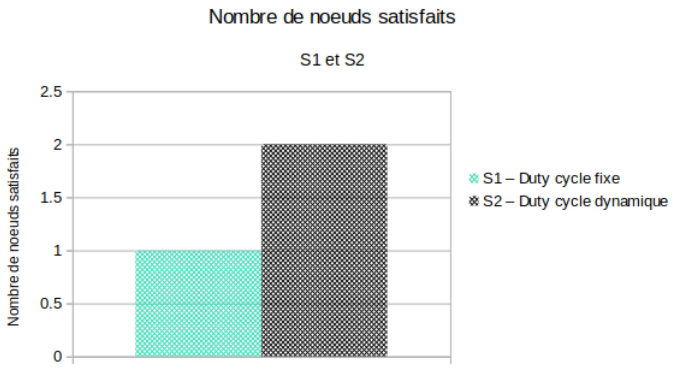


FIGURE IV.8 – Le nombre de noeuds satisfaits S1-S2

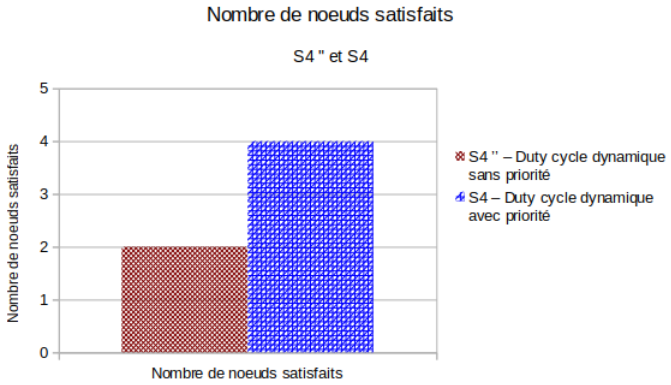


FIGURE IV.9 – Le nombre de noeuds satisfaits S4-S4//

D'après les résultats obtenus, nous constatons que l'approche de DC dynamique a permis d'augmenter le nombre de noeuds satisfaits dans les différents scénarios testés grâce à l'emprunt d'un temps supplémentaire, au dépit d'une augmentation du ToA. Cette augmentation de ToA revient à l'utilisation de nouveaux champs pour la manifestation des informations du noeud (type, durée).

La Table IV.1 montre la différence en termes de taille et de ToA des trames impliquées dans le mécanisme de partage de duty cycle. La longueur du message « Join Request » est augmentée de 6 octets. Le ToA du « Join Request » (demande de jointure de base + phase d'enregistrement) est augmenté d'environ 0,166 sec en moyenne dans l'approche proposée. Cette augmentation considère le cas extrême dans nos expérimentations (*mode*1) et reste tout de même ajustable pour une utilisation pratique (dépendamment du mode de configuration utilisé pour la transmission de la trame impliquant des données de partage du duty cycle). En outre, notre mécanisme n'a pas introduit de nouveaux messages, mais s'est basé sur les messages LoRa (Join, message confirmé) en exploitant des champs d'utilisations futures (RFU) et les commandes MAC.

Table IV.1. La taille du paquet « Join Request » et son ToA

	Taille du Join Request	ToA du Join Request
DC fixe	18 Bytes	1.48 sec
DC dynamique	24 Bytes	1.646 sec

Notre approche dynamique maximise le nombre de noeuds satisfaits. De plus, la phase d'enregistrement des types de noeuds avant la transmission augmente la possibilité d'emprunter du temps supplémentaire, indépendamment du début de l'activité des donneurs, mais plutôt selon leurs disponibilité pendant un cycle d'1 heure. Par exemple, lorsque les noeuds manifestent leurs types et durées pendant leur première transmission, les donneurs risquent de se manifester tardivement par rapport aux demandeurs. Par ailleurs, le fait qu'un noeud persiste à exprimer sa demande tant qu'il a la possibilité d'envoyer ses paquets lui donne plus de chance d'être satisfait avant l'expiration de son duty cycle. Dans la cas contraire sa transmission s'arrête sans bénéficier d'un temps supplémentaire.

En outre, en plus d'une dynamicité dans le temps d'activité des noeuds enregistrés dans le réseau LoRaWAN, notre approche permet aux noeuds prioritaires d'être satisfaits en premier grâce à une classification selon des critères différents, par exemple, les applications urgentes ou les noeuds ayant un faible niveau de batterie.

Table IV.2. PLR des demandeurs pour S3 et S4 (dû au dépassement de DC)

	D_1	D_2	D_3
DC fixe	42%	62.5%	14%
DC dynamique	0%	50%	0%

Les résultats obtenus en termes de PLR sont présentés dans la Table IV.2. Nous constatons que le mécanisme de DC dynamique utilisant une stratégie d'allocation du temps additionnel nous permet une atténuation du PLR d'environs 35.27% par rapport au DC fixe. En effet, dans le cas du DC fixe, D_1 interrompt sa transmission arrivant à son $116^{ème}$ paquet. D_2 interrompt sa transmission arrivant à la transmission de son $4^{ème}$ paquet. Tandis que D_3 arrête la transmission de ses paquets arrivant au $43^{ème}$ paquet. Alors que l'utilisation du DC dynamique dans S3 et S4 permet à D_1 et D_3 de finir toutes leurs transmissions. Cependant, pour cause d'insuffisance du temps donné, D_2 ne pourra pas continuer la transmission de tous ses paquets.

En résumé, notre proposition rend l'utilisation du duty cycle plus efficace en gérant d'une manière dynamique les différents temps autorisés aux différents noeuds afin de maximiser le nombre de noeuds satisfaits. Le mécanisme proposé maximise l'utilisation du canal en profitant des périodes d'inactivité de certains noeuds (donneurs) au profit des noeuds qui en ont besoin (demandeurs). A la différence de la solution adoptée dans [88], aucune privation ne peut être notée à cause de cette allocation de ressources qui peut être plus efficace en privilégiant les noeuds qui devraient finir leurs transmissions le plus tôt possible (ToA le plus faible, Batterie plus faible, etc.).

Cela résout le problème des noeuds qui veulent dépasser leur limite ToA pour les demandes urgentes, par exemple. Grâce à cet algorithme, nous pouvons maximiser le nombre de noeuds satisfaits qui doivent occasionnellement dépasser leur limite de DC. Notre solution a été mise en œuvre et testée de manière approfondie. Selon les résultats expérimentaux, nous avons montré que le mécanisme de partage proposé, avec une sélection appropriée des demandeurs en fonction des contrats de service (priorités), améliore la qualité globale du réseau LoRaWAN. Notre mécanisme comprend également le partage en mode statique (phase d'enregistrement) et en situation dynamique pendant la transmission (dans les paquets) du nœud pour maximiser

les chances de transmission d'un noeud en cours de modification de l'environnement (SF plus grand) ou rafales de données urgentes, etc. Notre étude s'est limitée à quelques noeuds pour valider la preuve de notre concept en situation réelle. Dans la section suivante, nous combinons le DC dynamique à l'E-ADR aussi bien pour des noeuds fixes que de noeuds mobiles.

2.4 Duty Cycle dynamique combiné à E-ADR pour de meilleures performances

Pour apporter de plus amples améliorations de performances, nous proposons de combiner nos deux contributions (E-ADR - DC dynamique) et d'évaluer cette combinaison dans deux contextes différents : Noeuds statiques et Noeuds mobiles. Pour se faire, nous apportons quelques modifications aux algorithmes présentés précédemment.

2.4.1 Description du fonctionnement de la combinaison

2.4.1.1 Définition des informations des noeuds (Type, T_{Dem}/T_{Don})

Les noeuds estiment le ToA nécessaire pour leurs transmissions (ToA_E) en se basant sur un cas extrême (le $mode1$), qui garantira la réception des paquets d'enregistrement, et qui évitera de priver les noeuds de leurs propres temps autorisé $36sec$ et d'effectuer un sur-prêt du temps considéré. Par exemple, si l'estimation du ToA_E est faite au début selon un SF plus petit et que plus tard le noeud aurait besoin d'un plus grand SF, alors ce dernier ne pourra plus récupérer le temps supplémentaire puisqu'il se pourrait qu'il soit déjà exploité par un autre noeud. Les noeuds définissent leurs types et leurs T_{Dem} ou T_{Don} selon la Figure IV.1. Ceci dit, le temps demandé ou à donner peut être ajusté en cours de transmission en fonction de l'adaptation du débit grâce à E-ADR.

2.4.1.2 Mise à jour des paramètres de partage

Lors de l'envoi d'un paquet P, le noeud vérifie si le ToA_{limit} lui permet de transmettre son prochain paquet utilisant le mode de configuration courant ($mode(i)$), sinon sa transmission est interrompue. Si son ToA_{limit} est suffisant, il calcule le temps à récupérer ($T_{écart}$) suite à l'envoi du paquet courant avec un mode $(i) > mode1$ selon l'Eq IV.3.

$$T_{écart} = (ToA_p(mode1)) - (ToA_p(mode(i))) \tag{IV.3}$$

Si le noeud est un demandeur qui n'a toujours pas été satisfait, ce dernier envoie son paquet en réduisant son T_{Dem} (T_{Dem}=T_{Dem}-$T_{écart}$), et il actualise son ToA_{limit} en réduisant le ToA consommé lors de la transmission du paquet ($ToA_p(mode(i))$). Si il s'agit d'un ancien demandeur (déjà satisfait), ce dernier devient donneur d'un temps T_{Don}=$T_{écart}$.

Si le noeud est un donneur ou était donneur auparavant, il envoie son paquet en actualisant son T_{Don} (T_{Don}=$T_{écart}$) et il met à jour son ToA_{limit}.

Sinon, il s'agit d'un noeud classique qui ne participe pas au partage. Ce dernier actualise seulement son ToA_{limit}.

L'algorithme présenté dans la Figure IV.10 permet la mise à jour du statut des noeuds lors de la modification du mode de configuration. Cela signifie, que dans le cas d'utilisation

d'un SF plus petit : (1) un ancien donneur reste donneur avec un temps supplémentaire (T_{Don}=T_{Don}+$T_{écart}$), (2) un demandeur peut rester demandeur, mais d'un temps plus faible que celui demandé auparavant ou bien (3) un ancien demandeur devient donneur d'un temps $T_{écart}$ qu'on lui a alloué avant, mais dont il n'a plus besoin après adaptation de la configuration.

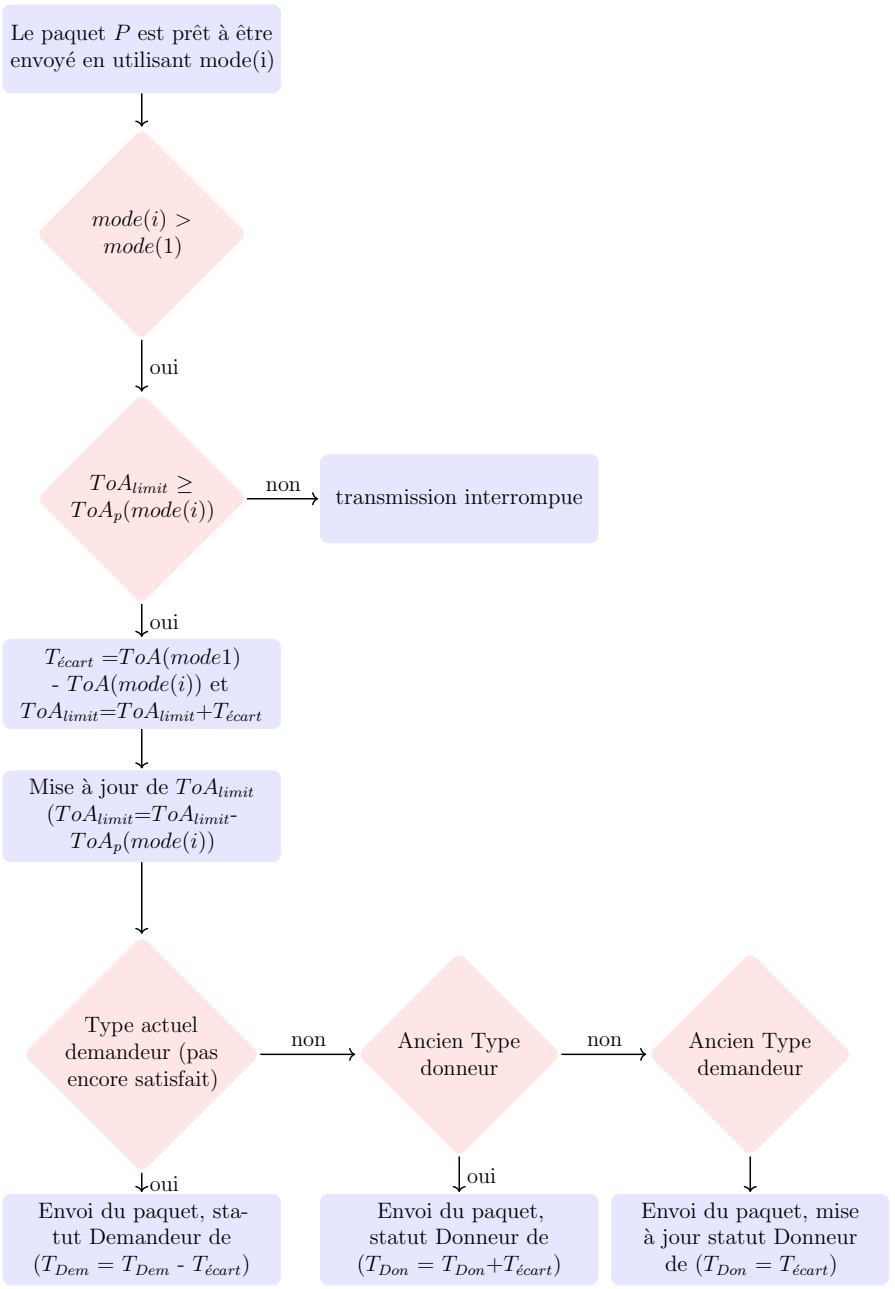


FIGURE IV.10 – Mise à jour de l'état de participation au DC d'un noeud

2.4.1.3 Traitement au niveau serveur pour envoyer un acquittement

Après réception des paquets, le serveur vérifie d'abord l'adaptation de débit selon la stratégie ADR utilisée par le protocole. Ensuite, le serveur devrait procéder à un traitement des

demandes de participation au partage de DC selon les indications contenues dans les paquets reçus. Ainsi l'acquittement envoyé par le serveur porte aussi bien la nouvelle configuration que l'accord ou non de la demande éventuelle.

2.4.2 Effet du DC Dynamique sur l'adaptation du débit dans un réseau de noeuds fixes

Nous nous intéressons dans cette section à évaluer l'apport de la combinaison du DC Dynamique avec les mécanismes d'adaptation de débit. Nous évaluons le DC dynamique, avec une politique d'allocation pour les noeuds ayant la plus faible demande avant, conjointement avec ADR basique et nous le comparons au cas d'utilisation de l'ADR Basique sans partage de DC.

2.4.2.1 ADR Basique

Dans ce scénario, D_1, D_2, et D_5 sont des noeuds dédiés à des applications de sécurité (transmettant 2700 Bytes chacun). D_3 est un capteur permettant d'identifier les étudiants (D_3 : 400 Bytes) et D_4 est un capteur de température (D_4 : 100 Bytes).

Les 5 noeuds transmettent leurs 3 premiers paquets ($n = 3$) en utilisant $mode1$ (SF12, BW125). La configuration des noeuds sera mise à jour après $n = 3$ paquets reçus. D_1 et D_2 utiliseront $mode2$ pour leurs prochains paquets, D_3 utilisera $mode4$, D_4 a fini sa transmission (100 Bytes envoyé en deux paquets), et D_5 utilisera $mode3$. Cette configuration ne changera pas puisque les noeuds sont fixes et nous supposons que les conditions sont stables (pas de grande variation de la force du signal RSSI).

D_1 et D_2 atteignent la limite ToA (ToA_{limit}=36sec) lors de la transmission de leurs $27^{ème}$ paquets. D_5 atteint son ToA_{limit} en arrivant à la transmission de la $40^{ème}$ paquet. D_3 et D_4 finissent leurs transmissions.

2.4.2.2 ADR Basique - DC dynamique

Nous reprenons le même scénario précédent. Cette fois-ci, le mécanisme de DC dynamique est activé. Les noeuds ajustent leurs ToAs (ToA_E) en utilisant à chaque fois le nouveau mode de configuration afin de définir leurs statuts. Le mode, utilisé comme configuration initiale, correspond au débit le plus faible (le pire cas) qui pourrait être utilisé lors de leurs transmissions. D_1, D_2, et D_5 s'annoncent demandeurs de 47.14 sec chacun. D_3 et D_4 s'annoncent donneurs de respectivement : 18.18 sec et 30.75 sec.

Une fois les 3 paquets sont reçus, le serveur alloue un nouveau mode de configuration aux 5 noeuds et vérifie les demandes des noeuds afin de les traiter. Il cumule le $T_{G_{Don}}$=48.93 sec et commence l'allocation du temps supplémentaire aux demandeurs selon une stratégie FIFO. Le serveur répond par un acquittement favorable à D_1, actualise son $T_{G_{Don}}$ ($T_{G_{Don}}$=1.79 sec) et répond par un acquittement défavorable aux demandeurs D_2 et D_5 puisque le $T_{G_{Don}}$ n'est pas suffisant pour les satisfaire.

Une fois les acquittements reçus par les noeuds, ces derniers actualisent leurs modes de configuration alloués, et leurs ToA_{limit} (pour les demandeurs satisfaits). Ils recalculent leurs ToA_E et actualisent leurs statuts. En effets, en utilisant $mode2$, D_1 devient donneur de 25.76 sec. D'autre part, D_2 et D_5, n'ayant pas été satisfaits au premier coup, deviennent demandeurs

de respectivement : 21.38 sec et 4.12 sec. Quant aux donneurs, D_3 utilisant *mode*4 donne une seconde fois un T_{Don}=7.3 sec, D_4 (100 Bytes envoyé en deux paquets) a fini sa transmission et est en veille. Les noeuds D_1, D_2, D_3 et D_5 continuent leurs transmissions.

En recevant le $4^{ème}$ paquet de chaque noeud, le serveur actualise sa table de demandeurs ainsi que les T_{Dem} correspondant à chaque demandeur, cumule le $T_{G_{Don}}$ ($T_{G_{Don}}$=33.06 sec) et commence l'allocation du temps additionnel. Le serveur commence par allouer le T_{Dem} demandé par D_5, actualise son $T_{G_{Don}}$ ($T_{G_{Don}}$=28.94 sec), et lui envoie un acquittement favorable. Ensuite, il alloue le T_{Dem} demandé par D_2 et actualise son $T_{G_{Don}}$ ($T_{G_{Don}}$=7.56 sec), et lui envoi un acquittement favorable.

D_2 et D_5 mettent à jour leurs ToA_{limit} et finissent leurs transmissions.

2.4.2.3 Évaluation des pertes

Nous présentons dans ce paragraphe le taux de perte (PLR) des différents noeuds demandeurs D_1, D_2, et D_3 lors de l'utilisation de "ADR Basique" et de "ADR Basique - DC" (Table IV.3).

Table IV.3. PLR des demandeurs (dû au dépassement de DC)

	D_1	D_2	D_3
ADR Basique	32.5%	32.5%	13.04%
ADR Basique - DC dynamique	0%	0%	0%

Nous remarquons que l'introduction du DC a permis la minimisation des pertes (PLR atténué) grâce au basculement de demandeurs à donneurs après les mise à jours de configurations, permettant de profiter de toute durée autorisée non utilisée par les noeuds donneurs. En effet, le temps d'activité des noeuds change selon la configuration allouée à chaque fois, et donc l'adaptation des paramètres de configuration a permis aux noeuds de réduire leurs ToAs et donc une réduction du temps demandé et une augmentation du temps à donner. Ceci a suscité une satisfaction complète des noeuds demandeurs avec atténuation du PLR (Table IV.3) et un temps à donner restant pouvant satisfaire plus de demandes en cas d'événement provoquant une nouvelle rafale de paquets.

2.4.3 Effet du DC Dynamique sur l'adaptation du débit dans un réseau de noeuds mobiles

Dans cette partie, on s'intéresse à l'évaluation de la combinaison « E-ADR - DC Dynamique » dans un contexte de mobilité. Nous considérons 5 noeuds au bord de 5 différents bus de l'école "Smart Bus" empruntant différentes trajectoires connues. Ces 5 noeuds servent à envoyer une donnée de 100 Bytes après chaque arrêt concernant la trajectoire parcourue, le dernier et le prochain arrêt, la vitesse, l'état du trafic routier, le temps d'arrivée estimé, et le statut des élèves (à bord ou pas) [89]. Le nombre des arrêts diffère d'un bus à un autre : 39 arrêts pour D_1, 30 arrêts pour D_2, 33 arrêts pour D_3, 24 arrêts pour D_4 et 21 arrêts pour D_5. Certains bus se déplaceront autour des emplacements des gateways et d'autres s'en éloigneront parfois.

2.4.3.1 E-ADR

Puisque les bus traversent des trajectoires connues, le mécanisme d'adaptation de débit utilisé est la première version « E-ADR » présentée dans le Chapitre 2. Les 5 noeuds dans les 5 différents bus transmettent leurs premiers $n = 3$ paquets en utilisant le $mode1$. Ils estiment le ToA de tous leurs paquets en utilisant le $mode1$ (ToA_E). Ensuite, après chaque réception de $n = 3$ paquets, le serveur estime le prochain mode de configuration pour les prochains $n = 3$ paquets selon la stratégie E-ADR (voir Chapitre 2, Section 3), et envoie la nouvelle configuration au noeud à travers un acquittement.

Contrairement au contexte fixe, la configuration des noeuds change fréquemment dans un contexte mobile. Pour cela, nous avons relevé tous les changements de mode qui ont eu lieu pendant l'expérimentation, les ToAs consommés pendant le cycle d'une heure ainsi que le PLR de chaque noeud suite à un dépassement de duty cycle (Table IV.4). Nous rappelons que les pertes dues aux mauvaises configurations sont atténuées grâce au mécanisme E-ADR.

Les noeuds D_4 et D_5 finissent leurs transmissions sans atteindre le ToA_{limit}=36 sec. Par contre, D_1, D_2, et D_3 atteignent le ToA_{limit} et interrompent leurs transmissions arrivant à leurs $24^{ème}$, $21^{ème}$ et $22^{ème}$ paquets.

Noeud	Les configurations utilisées	ToA consommé
D_1	mode1, mode4, mode5, mode3, mode2, mode1, mode1, mode2	36 sec
D_2	mode1, mode5, mode6, mode4, mode2, mode1, mode2, mode1	36 sec
D_3	mode1, mode2, mode4, mode3, mode1, mode2, mode1, mode2	36 sec
D_4	mode1, mode1, mode2, mode5, mode3, mode4, mode7, mode8	13.87 sec
D_5	mode1, mode3, mode5, mode2, mode3, mode5, mode7	14.06 sec

Table IV.4. Allocation des modes, consommation du ToA et PLR

2.4.3.2 E-ADR - DC dynamique

Nous reprenons le même scénario de bus scolaire et nous intégrons le mécanisme de partage de duty cycle. Au début, les noeuds estiment leurs ToA_E selon le $mode1$ et définissent leurs types et le temps à donner ou demander. Ils s'annoncent tous demandeurs de respectivement : 66.55sec (D_1), 42.88 sec (D_2), 50.77 sec (D_3), 27.11 sec (D_4), et 19.22 sec (D_5). Les noeuds envoient leurs statuts au serveur à travers la première transmission.

Après $n = 3$ transmissions, le serveur estime le prochain mode pour chaque noeud, vérifie son $T_{G_{Don}}$ ($T_{G_{Don}}$=0) et envoie un acquittement pour chaque noeud indiquant le nouveau mode de configuration et que la demande de temps additionnel n'est pas satisfaite.

Les noeuds, en recevant les acquittements, actualisent leurs modes de configuration, vérifient la réponse à leurs demandes, vérifient si leurs temps autorisés leur permettent la transmission du prochain paquet en utilisant le nouveau mode de configuration, et ils calculent l'écart $T_{écart}$. Cette fois-ci les noeuds s'annoncent aux serveurs également en tant que demandeurs de : 60.39sec (D_1), 35.98 sec (D_2), 46.34 sec (D_3), 27.11 sec (D_4), et 13.31 sec (D_5). En effet, après la mise à jour de la configuration des $n = 3$ prochains paquets (plus faible SF), les T_{Dem} sont réduits.

Les noeuds continuent leurs transmissions tant que leurs temps autorisés ToA_{limit} le permettent et actualisent leurs modes de configurations alloués par le serveur après chaque $n = 3$. Au fur et à mesure de l'adaptation de débit suite au déplacement des noeuds, leurs T_{Dem} sont décrémentés, et certains noeuds deviennent donneurs. En effet, les noeuds D_4 et D_5, arrivés à leurs $12^{ème}$ et $9^{ème}$ paquets, respectivement, s'annoncent donneurs au serveur durant chaque paquet envoyé et à chaque fois d'un T_{Don} équivalant à l'écart $T_{écart}$ calculé lors de la modification du mode de configuration (Voir Figure IV.10). Le serveur cumule les T_{Don} dans le $T_{G_{Don}}$, vérifie à chaque fois si ce dernier est suffisant pour une allocation d'un T_{Dem}. Les demandeurs D_2 et D_3 recevront un acquittement positif (demande autorisée) après l'envoie de leurs $18^{ème}$ et $15^{ème}$ paquets, leur permettant de poursuivre leurs transmissions au-delà de 36 sec.

D_2 et D_3 actualisent leurs ToA_{limit} après l'acceptation de partage, leurs modes de configuration et continuent à calculer un $T_{écart}$ avant l'envoi de chaque $n = 3$ paquets. Ces derniers passent de demandeurs à donneurs d'un T_{Don} à partir de leurs $21^{ème}$ et $19^{ème}$ envoi.

En cumulant les T_{Don} de D_2 et D_3, le serveur arrive à satisfaire D_1 en arrivant à son $20^{ème}$ paquet.

2.4.3.3 Évaluation des pertes

Nous présentons dans cette partie le taux de perte (PLR) des différents noeuds demandeurs D_1, D_2, et D_3 lors de l'utilisation de "E-ADR" et de "E-ADR - DC dynamique".

Table IV.5. PLR des demandeurs (dû au dépassement de DC)

	D_1	D_2	D_3
E-ADR	38.46%	30%	33.33%
E-ADR - DC dynamique	0%	0%	0%

La Table IV.5 présente les PLRs obtenus dans les deux cas. En effet, bien que l'E-ADR soit le plus performant par rapport aux variantes ADR présentées dans le Chapitre 2, ce dernier minimiserait les pertes, mais sans les atténuer complètement lors de la transmission de quantité de données assez importante en se déplaçant loin des gateways. C'est dans ce contexte que nous avons proposé d'introduire l'approche de duty cycle dynamique combinée à l'E-ADR. La Table IV.5 permet de voir l'atténuation du PLR suite à l'introduction de mécanisme de DC dynamique. La mise à jour fréquente de la configuration ainsi que celle des statuts des noeuds (demandeur/donneur) a permis de bénéficier d'un partage de temps additionnel entre les noeuds, aidant les demandeurs à terminer leurs transmissions et éliminer les pertes dues au dépassement de duty cycle dans ce scénario.

En résumé, au même temps que la mise à jour de la configuration des noeuds, le ToA des noeuds est actualisé. Le temps demandé par les demandeurs est diminué après chaque adaptation de configuration tandis que le temps donné par les donneurs augmente. D'autant plus, certains demandeurs basculent en des « donneurs » après plusieurs mises à jour (D_2, et D_3), ce qui nous permet de conclure que la combinaison de l'E-ADR au mécanisme de partage de DC fournit une meilleure fiabilité de transmissions, en minimisant la probabilité de perte de paquets face à la transmission de quantité de données importante.

Nous avons montré dans ce qui précède que l'approche de partage de duty cycle dynamique associé à la solution d'adaptation de débit permet de diminuer, et même d'atténuer dans certains cas le PLR. Toutefois, même si l'allocation du SF est adéquate aux besoins des noeuds, la gestion de la distribution de ces SF sur les canaux LoRaWAN n'est pas considérée. Une utilisation d'un même SF sur le même canal provoquerait une augmentation significative de la probabilité de collisions dans le cas d'un réseau LoRaWAN dense (plusieurs noeuds LoRa). C'est dans ce contexte que nous proposons dans la seconde partie de ce chapitre une nouvelle stratégie d'accès aux ressources LoRaWAN.

3 – Approche d'allocation triplement conjointe SF-Slot-Canal

LoRaWAN a été conçu à l'origine pour être utilisé comme un réseau en étoile à grande échelle. En tant que tel, il est donc difficile pour les nœuds de découvrir leurs voisins (d'autres nœuds) dans le réseau, sachant que sans aucune information concernant les voisins ou leur calendrier de transmission, le risque de créer des collisions est non négligeable [53]. En fait, même si la quantité de données générées par chaque nœud est faible, le grand nombre de noeuds essayant d'accéder au même canal sans fil en même temps peut être ingérable par des techniques telles que P-ALOHA. En outre, les transmissions simultanées utilisant le même canal et la même configuration entraînent une perte de tous les paquets (collision), sauf si l'une de ces transmissions utilise une TP beaucoup plus élevée conformément à la condition SNR (le signal est reçu au moins 6 dB au-dessus des autres signaux) [36]. Celle-ci éliminera toutes les autres transmissions et sera la seule à être reçue. Ce phénomène est appelé « effet de capture » et a été étudié dans [37].

D'autre part, la méthode de sélection des canaux de transmission dans le protocole LoRaWAN est un autre facteur important impactant sur la performance du réseau. Les canaux de transmission sont sélectionnés de manière pseudo-aléatoire, et c'est ce qui fait la forte probabilité que des transmissions simultanées avec le même SF utilisent le même canal, induisant des pertes de paquets et une dégradation de la QoS.

Ces limitations deviennent un sérieux goulot d'étranglement pour la performance des réseaux basés sur LoRaWAN dans la réalisation des déploiements à grande échelle, tels que dans l'agriculture (ferme connectée), des centaines de nœuds sont envisagés pour augmenter la production en surveillant les conditions climatiques et en contrôlant à distance les systèmes d'irrigation et d'éclairage. Ces noeuds utilisent le service de classe A pour envoyer leur données. Rappelons qu'un noeud de classe A transmet d'abord les données UL à la gateway et ouvre ensuite jusqu'à deux fenêtres de réception (RX1 et RX2), donnant la possibilité à la gateway d'envoyer une transmission DL. Le slot de transmission est programmé par le noeud sur la base des exigences de sa propre trame UL. Dans ce schéma de communication basé sur P-ALOHA, d'une part, le noeud transmet son paquet lorsqu'il est prêt sans écouter le canal générant ainsi des collisions et des taux de perte de paquets élevés. D'autre part, la gateway n'a aucun contrôle sur le noeud, et toute communication DL doit se faire forcément dans les deux fenêtres suivant une transmission UL, ce qui augmente la latence sans oublier le risque de collision avec d'autres transmissions UL. En dépit des restrictions imposées par les autorités de régulation (Duty Cycle), le problème de collision se pose toujours.

Dans ce cadre, nous proposons une solution plus performante que P-ALOHA permettant

d'éviter les collisions à travers une planification des ressources de transmission pour tous les paquets assurant le choix du slot, du canal de transmission et des paramètres de configuration selon les exigences des applications mais aussi des ressources disponibles en adoptant une stratégie de priorisation de l'allocation des ressources.

Nous considérons dans notre approche des capteurs transmettant des données périodiques la plupart du temps, mais pouvant faire face à des déclenchements d'événements nécessitant l'envoi de données supplémentaires. L'idée principale est basée sur la technique d'accès TDMA pour la réservation des slots, sachant que cette technique fonctionne lorsque le serveur possède une vue globale sur le réseau sans considérer les irrégularités induisant une augmentation des collisions. Pour résoudre cet inconvénient et considérer les éventuelles irrégularités, nous choisissons d'ajouter une dynamicité dans la programmation des slots. Les noeuds pourront alerter le serveur lorsque ces derniers détectent plus de données à transmettre de ce qui a été prévu initialement (rafale de données suite à un évènement, par exemple). De plus, cette dynamicité permettra de tolérer les environnements mobiles, où les besoins d'un noeud mobile changent en fonction du déplacement. Cette allocation de slot est faite conjointement avec une stratégie de sélection de canaux ayant pour but d'éliminer la probabilité d'utilisation du même canal avec le même SF au même temps. Ceci en créant des tables indiquant la disponibilité des canaux pour chaque mode de configuration. Cette approche doublement conjointe sera associée à la solution E-ADR proposée précédemment pour être finalement "Triplement conjointe SF-Slot-Canal". Le serveur sera responsable de : l'allocation des paramètres de configuration, l'allocation des slots selon les besoins des noeuds et la priorité des applications ainsi que l'allocation des canaux, de manière dynamique après chaque réception de paquet.

Cette nouvelle approche permettra d'allouer des ressources et planifier le trafic des différents noeuds dans le temps et sur l'ensemble des canaux disponibles, réduisant au maximum le taux de collisions et les pertes de données. Notre proposition a été mise en oeuvre à l'aide des modules LoRa Waspmote SX1272 [49, 83], STM32 [84] et Pycom Lopy4 [90] afin de valider son efficacité en comparant les combinaisons : "E-ADR ALOHA" et "E-ADR SF-Slot-Canal".

3.1 Mécanismes d'accès proposés pour LoRaWAN : état de l'art

De nouvelles propositions sont apparues ces dernières années pour remplacer le P-ALOHA utilisé par défaut dans LoRaWAN par d'autres mécanismes d'accès plus adaptés aux applications IoT. Les propositions sont en général motivées par l'un des deux objectifs suivants : le premier est d'améliorer l'extensibilité et la fiabilité des communications, ce qui, à son tour, augmente l'efficacité énergétique et améliore la capacité de la gateway à desservir plus de dispositifs en réduisant les collisions. Le deuxième facteur motivant la mise à niveau de l'accès aux canaux LoRaWAN est de prendre en charge des cas d'utilisation avec des exigences de QoS prédéfinies, comme les applications de contrôle et de sécurité. Cela devrait conduire à des solutions d'accès basées sur la réservation des ressources et l'attribution de fréquences, de temps et de SF au même temps, ce qui n'a pas été considéré dans LoRaWAN. Nous résumons ci-dessous les améliorations de l'accès aux canaux spécifiques au protocole LoRaWAN qui sont proposées dans la littérature.

Pour commencer, plusieurs études ont évalué la technique d'accès P-ALOHA utilisée dans le protocole LoRaWAN standard en utilisant des simulations tout en prenant en compte les informations de la couche physique et de la couche de liaison en termes de latence, de

fiabilité et de débit [53, 57, 91, 92]. La plupart de ces études concluent que bien que P-ALOHA fonctionne bien avec de faibles charges de trafic, cette technique souffre de congestion du trafic sur la liaison montante (UL) en raison de l'impossibilité de vérifier si le canal est occupé avant de transmettre. De plus, le protocole P-ALOHA pourrait générer de grandes inefficacités lorsque le nombre de nœuds d'un réseau augmente de manière significative.

Dans ce cadre, des solutions ont été récemment proposées dans la littérature pour envisager une approche Time-slotted. Dans un premier temps, le Slotted-ALOHA (S-ALOHA) a été utilisé dans le protocole LoRaWAN dans de nombreux travaux de recherche [93–95]. Ensuite, de nouveaux algorithmes basés sur l'approche Time-slotted ont été proposé [96–104].

Le protocole d'accès Slotted-ALOHA est une variante du protocole P-ALOHA et est largement utilisé dans les communications locales sans fil. S-ALOHA contrairement à P-ALOHA dispose d'un canal divisé en slots. Les utilisateurs ne peuvent commencer à transmettre que sur les débuts des slots [93]. Chaque slot est constitué de deux parties : le temps de transmission (ToA) et les fenêtres de réception (T_{RX}). Si deux ou plusieurs nœuds transmettent leurs paquets en même temps, ils se chevauchent complètement au lieu de se chevaucher partiellement. Ainsi, seule une fraction des slots dans lesquels le paquet entre en collision est programmée pour être retransmise. Cela double presque l'efficacité du S-ALOHA par rapport à P-ALOHA pur [94]. L'introduction de S-ALOHA a contribué dans l'élimination partielle des collisions, car chaque nœud est autorisé à envoyer un paquet uniquement au début d'un slot. Cependant, l'accès au support reste incontrôlé. L'apparition d'une collision dépend de la décision de plusieurs noeuds de transmettre un paquet simultanément au début du même slot sur le même canal et avec le même SF [95]. Le manque de coordination et l'absence de programmation dans les transmissions de paquets sont les principales raisons qui expliquent les mauvaises performances en temps réel de P-ALOHA et S-ALOHA.

Afin de réduire les inconvénients du S-ALOHA, [96] a proposé un algorithme de planification centralisée des slots visant l'amélioration de l'extensibilité de LoRaWAN mise en oeuvre du côté noeud. Cette programmation de slots n'est exploitée que par les nœuds de classe B, car la synchronisation repose sur des Beacons de classe B. L'énergie consommée liée à la diffusion des slots par des capteurs additionnels est négligée, ce qui remet en question l'applicabilité de l'étude. De plus, le cas où ces slots sont calculés par les noeuds, sachant que la capacité de mémoire et la puissance de calcul des noeuds LoRa est limitée, est une contrainte pour cette approche dans un environnement réel [97].

Récemment, un effort a été fait dans [98] pour accroître l'extensibilité des réseaux LoRa divisant la largeur de bande disponible en un seul canal descendant synchrone et plusieurs canaux asynchrones de liaison montante/descendante. La gateway envoie un beacon pour synchroniser les nœuds, il transporte également des informations précisant la puissance de transmission et les facteurs d'étalement autorisés à utiliser par des nœuds pour la transmission en liaison montante. Au réveil, les nœuds écoutent la dernière balise pour synchroniser l'information et transmettent ensuite les données de manière aléatoire (ALOHA). Bien que cette stratégie « MAC RS-LoRa » proposée dans [98] atténue les collisions de paquets de près de 20 % par rapport au LoRaWAN standard, elle ne les élimine pas, car les messages de la liaison montante pourraient encore entrer en collision avec les beacons des passerelles voisines. De plus, le scénario d'applications prioritaires, où les paquets ne peuvent attendre un Beacon pour être transmises (cas d'urgence), n'est pas considéré, mais aussi ce comportement étant identique à la classe B réduit l'efficacité énergétique.

Dans [99], le processus de synchronisation est lancé par les nœuds, ensuite la gateway calcule de manière centralisée le programme complet (slots) pour chaque nœud. Ici, le programme est calculé en fonction des exigences de l'application, telles que la périodicité des données, et il est renvoyé aux nœuds en utilisant une structure de données probabiliste spécifique, appelée « filtre Bloom ». En raison de la nature probabiliste de ces structures, plusieurs nœuds partagent le même slot avec une certaine probabilité et, par conséquent, cette technique n'élimine donc pas les collisions. De plus, ces approches n'ont pas été vérifiées expérimentalement.

Dans [100], des dispositifs de localisation (GPS pour l'extérieur et UWB pour l'intérieur) sont utilisés avec les nœuds LoRaWAN pour obtenir de manière opportuniste une synchronisation temporelle, utilisée pour mettre en œuvre une stratégie efficace d'accès programmé au canal. Cette technique améliore le débit par rapport au LoRaWAN standard, mais nécessite du matériel supplémentaire coûteux et présente une consommation d'énergie plus élevée.

Les auteurs dans [101] envisagent une approche « Time-slotted (TS- LoRa) » pour permettre la collecte de données en masse dans LoRaWAN. Dans ce travail, une phase d'overhead est nécessaire pour la procédure de jointure et la synchronisation avant chaque collecte de données. Il en résulte des améliorations significatives en termes de PRR et de durée de vie des noeuds. Les auteurs dans [102] proposent également une approche centralisée pour le même problème. Les deux derniers travaux sont sans collisions, mais ne permettent pas la collecte de données en temps réel.

Une autre catégorie de travaux de recherche est l'utilisation de la technique TDMA dans le protocole LoRaWAN. Le principe est de diviser le canal en plusieurs slots.

En TDMA, chaque nœud se voit tout d'abord attribué un ou plusieurs slots pour sa transmission. Le nombre de slots de temps est déterminé par le nombre de trames à envoyer par chaque noeud. Les slots sont attribués aux paires d'émetteurs/récepteurs selon une certaine méthode d'attribution, permettant aux nœuds d'accéder au canal sans fil en utilisant le mécanisme TDMA [105–107]. Étant donné qu'un slot est attribué de manière unique à une paire d'émetteurs/récepteurs, le protocole d'accès TDMA permet une transmission de paquets sans collision, quelle que soit la charge de trafic, garantissant ainsi une grande efficacité de transmission et un faible délai de transmission grâce à l'élimination des retransmissions. Les auteurs dans [103] présentent une approche basée sur le protocole de communication TDMA asynchrone qui combine les capacités des radios de réveil (wake-up radios) à courte portée avec la connectivité à longue portée de LoRa, améliorant ainsi efficacement l'efficacité énergétique et la latence du réseau LoRaWAN. On-demand LoRa, proposée par ces auteurs, est une couche MAC, alternative à LoRaWAN, qui utilise deux stratégies TDMA différentes, appelées respectivement Unicast et Broadcast TDMA. Chaque noeud est équipé d'un émetteur-récepteur à faible puissance, conforme à la norme radio Wake-up, qui est normalement maintenue en état d'écoute profonde. Avec la technologie Unicast TDMA, la gateway envoie un Beacon de réveil à un noeud spécifique qui lance les communications en UL. Avec la technologie Broadcast TDMA, la gateway envoie un Beacon de réveil à plusieurs services d'urgence qui lancent des communications en UL en différé en utilisant des slots programmés. Bien que cette proposition réduit les collisions, l'utilisation de Beacon ne garantie pas une efficacité énergétique. De plus, cette solution nécessite un émetteur-récepteur sur le noeud LoRa non-standard qui reste en état d'écoute augmentant le coût et la consommation énergétique. En outre, les données non-périodiques en cas de détection d'évènement ne sont pas prises en considération

et ne sont pas supportées par le TDMA [104]. Même si TDMA réduit les collisions, cette technique ne permet pas de les éliminer, car même en l'utilisant dans le protocole LoRaWAN, la sélection des canaux reste pseudo-aléatoire provoquant ainsi des interférences. Sans une stratégie de sélection de canaux efficace et une certaines dynamicité pour considérer les éventuels changements, TDMA n'élimine pas les collisions.

La dernière catégorie de solutions pour améliorer l'utilisation de la bande passante est la procédure d'écoute avant transmission (LBT) [35, 108] pour les noeuds de classe A. Le mécanisme LBT permet à plusieurs noeuds de partager le même canal en surveillant continuellement le canal de manière à ne transmettre que lorsqu'il est disponible. Les émetteurs-récepteurs LoRa offrent un mode spécial appelé détection de l'activité des canaux (CAD). En cas de détection d'un préambule, l'émetteur se met en veille pendant une période de temps aléatoire, puis effectue à nouveau la détection des canaux. Bien que LBT permette une haute prévention des collisions, il augmente la puissance overhead requise par les nœuds pour la détection des canaux, un facteur crucial pour les nœuds de capteurs alimentés par batterie. En outre, la LBT contribue également aux retards de paquets dus au mécanisme back-off exponentiel.

Motivés par ce qui précède, nous nous sommes intéressés aux stratégies d'allocation de slots de transmissions. C'est dans ce contexte que nous proposons une allocation de slots dynamique basée sur la technique TDMA dynamique pour les Slots, la technique E-ADR pour le SF et la technique FIFO pour les canaux. L'idée est de mettre à jour l'allocation pendant la phase de transmission d'une manière dynamique afin de s'adapter aux :

— Cas de noeuds irréguliers ayant besoin de transmettre plus de données dans le cas de détection d'événements, ceci dit qu'ils auront besoin de SF-Slot-Canal supplémentaires, et

— Cas de mobilité où les besoins du noeud changent en fonction de ses déplacements et varient d'un instant à un autre.

Contrairement à la technique TDMA, TDMA dynamique permettra non seulement l'allocation de slots selon les SFs alloués, mais aussi la programmation de nouveaux slots pour la transmission des données irrégulières au lieu de l'utilisation de la technique P-ALOHA pour l'envoi de ces dernières. Notre approche, appelée « une allocation dynamique triplement conjointe SF-Slot-Canal », sera mise en oeuvre sur des modules LoRa et testée dans un scénario expérimental réel. La section suivante détaillera notre proposition.

3.2 Allocation dynamique triplement conjointe

Notre modèle permet au serveur d'allouer des ressources de transmission (canal et slot) pour les transmissions régulières ainsi que les transmissions irrégulières (supplémentaires) à travers une allocation dynamique après chaque n paquets. Les canaux sélectionnés pour ces transmissions ne doivent pas être utilisés pendant une durée d'un slot (durée de transmission du paquet en question). Cependant, les demandes des noeuds peuvent changer en raison de la mobilité ou de présence d'obstacles. Dans ce cas, le serveur mettra à jour cette allocation dynamiquement et fréquemment afin de réserver plus de ressources de transmission lorsqu'un noeud manifeste la présence d'une rafale supplémentaire de données, en supposant que

cette rafale de données pourrait être transmise ultérieurement. Cette mise à jour sera faite dynamiquement grâce à une vérification des requêtes des noeuds après la réception de chaque paquet ($n = 1$). Ainsi, les informations sur la nouvelle allocation déterministe (SF-Slot-Canal) seront envoyées dans les paquets ACK_{config} (utilisés pour l'adaptation de débit précédemment). L'objectif de cette approche dynamique est d'éliminer les collisions de la manière la plus efficace sur le plan énergétique et avec le moins de latence possible.

Le protocole LoRaWAN que nous proposons en plus de l'allocation du canal et du slot implémente la solution VHMM-based E-ADR pour la sélection de SF développée précédemment. Ce protocole, amélioré par nos solutions, sera nommé dans la suite du manuscrit « E-LoRaWAN ». Ils peuvent être fixes ou mobiles. Dans ce qui suit, nous détaillons le fonctionnement de notre protocole.

3.2.1 Principe d'allocation Slot/Canal

Dans le cas de P-ALOHA, même avec la procédure E-ADR, le serveur peut faire face à un nombre de noeuds, assignés le même mode de configuration (μ), sur le même canal. Dans LoRaWAN, le choix du canal est fait par le noeud de façon aléatoire et pas par le serveur. En ajoutant seulement E-ADR, le problème d'allocation de Slot/Canal n'est pas résolu. L'attribution du même mode de configuration sur le même canal à tous ces noeuds engendrera un taux de collision important (PLR très élevé) menant à l'augmentation du nombre de retransmissions induisant ainsi une augmentation rapide du ToA, franchissant aussitôt la limite du Duty Cycle (DC). C'est pour cela que nous proposons une approche de programmation des slots de transmission selon les demandes des noeuds tout en considérant la disponibilité des canaux. Le serveur attribuera soit des slots différés sur le même canal ou bien le même slot sur différents canaux.

Pour que cette allocation puisse avoir lieu, la synchronisation entre le noeud et le serveur sont effectuées à travers les commandes $AppTimeReq$ et $AppTimeAns$ au début ou pendant le cycle [109].

Figure IV.11 donne l'algorithme d'allocation de slot/canal pour le serveur. Le serveur dispose d'une table des N_{Ch} canaux LoRa à utiliser pour chaque mode de configuration ($mode\mu$). Chaque table ($T_{Chmode\mu}$) contient des informations sur les slots (début et fin du slot) attribués aux noeuds utilisant le $mode\mu$ et sur quel canal. Ces tables seront remplies au fur et à mesure des allocations. Ces tables seront exploitées pour la recherche des canaux libre.

À partir des identifiants ($DevEUI$ et $AppEUI$) de chaque noeud reçu dans le paquet d'enregistrement (Join Request), le serveur détermine le nombre de noeuds enregistrés dans le réseau (n_D), le nombre total des paquets programmés à la transmission (N_{Tj}) ainsi que la fréquence d'envoi ($P_{sendingj}$) de chaque noeud (j).

Après chaque n paquets reçus, le serveur estime le mode de configuration ($mode\mu(j)$) à allouer à chaque noeud j pour l'envoi de ses prochains n paquets. Il vérifie à travers l'indication envoyée par le noeud dans l'entête du $n^{ème}$ paquet si des données supplémentaires (k) ont été déclenchées ou pas. Si il existe k données supplémentaires irrégulières en attente de transmission, le serveur met à jour le nombre de paquets du noeud (N_{Tj}) afin de prendre en considération la programmation des slots pour leur transmission.

Le serveur calcule les n slots pour les n prochains paquets $[t_{(i,j)start}, t_{(i,j)end}]$ $(i = 1...n)$ selon les Équations (IV.4 et IV.5) pour les prochains n paquets du noeud j utilisant $mode\mu(j)$.

$$t_{(i,j)start} = t_{(i-1,j)end} + P_{sending(j)} \tag{IV.4}$$

$$t_{(i,j)end} = t_{(i,j)start} + ToA_{(i,j)_{mode\mu}} + t_{RX} \tag{IV.5}$$

En se basant sur le mode estimé ($mode\mu(j)$) et les slots calculés ($[t_{(i,j)start},\ t_{(i,j)end}]$) pour les n prochains paquets, le serveur recherche dans la table $T_{Chmode\mu}$ pour chaque slot i calculé ($i = 1...n$), le canal libre (Algorithme 3). Si le serveur trouve un canal $C_{lmode\mu}$ (avec $l = 1....N_{Ch}$) libre, il l'affecte au slot et actualise la Table $T_{Chmode\mu}$ en mettant le canal sélectionné $C_{lmode\mu}$ en indisponible jusqu'à $t_{(i,j)end}$. Sinon, le serveur recherche le premier prochain canal libre dans la table en se basant sur les différents t_{end} des différents slots enregistrés précédemment dans la table $T_{Chmode\mu}$ (Algorithme 4). Le slot de transmission ($[t_{(i,j)start},\ t_{(i,j)end}]$) sera décalé selon le premier prochain canal disponible, en supposant que l'intervalle de décalage n'influence pas l'allocation de configuration.

Ces informations seront envoyées dans l'acquittement en utilisant une nouvelle commande MAC avec CID=$0x1B$ appelée *NewChannelSlot* dans le champs *FOpts* (Voir Chapitre 1, Section 3.4.6.1, Figure I.10) [21].

En recevant les mises à jour du serveur, les noeuds mettent à jour leurs modes de configuration, leurs informations sur les slots et les canaux à utiliser pour leurs n prochaines transmissions.

Algorithme 3 : Recherche de canal disponible pour la transmission pendant τ utilisant le $mode\mu$ pour un noeud j

input : τ=$[t_{(i,j)start},\ t_{(i,j)end}]$: slot calculé pour le paquet i du noeud j ; $mode\mu$: le mode sélectionné ; $T_{chmode\mu}$: table des slots et canaux pour le mode μ ; N_{ch} : nombre de canaux par mode

output : C_l (Canal libre pendant le slot sélectionné)

```
for v = τ do
    for w ← 1 to N_Ch do
        X=T_Chmodeμ[v, w]
        if X==0 then
            C_l = w ; mettre à jour T_Chmodeμ[v, w] (T_Chmodeμ[v, w]==1) ; exit
        else if X≠ 0 then
            w = w + 1
        end
    end
    if C_l ≠ 0 then
        Affecter C_l au noeud j pendant le slot τ
    else if C_l == 0 then
        Chercher le premier canal libre pour le slot libre le plus proche (Voir Algorithme 2)
```

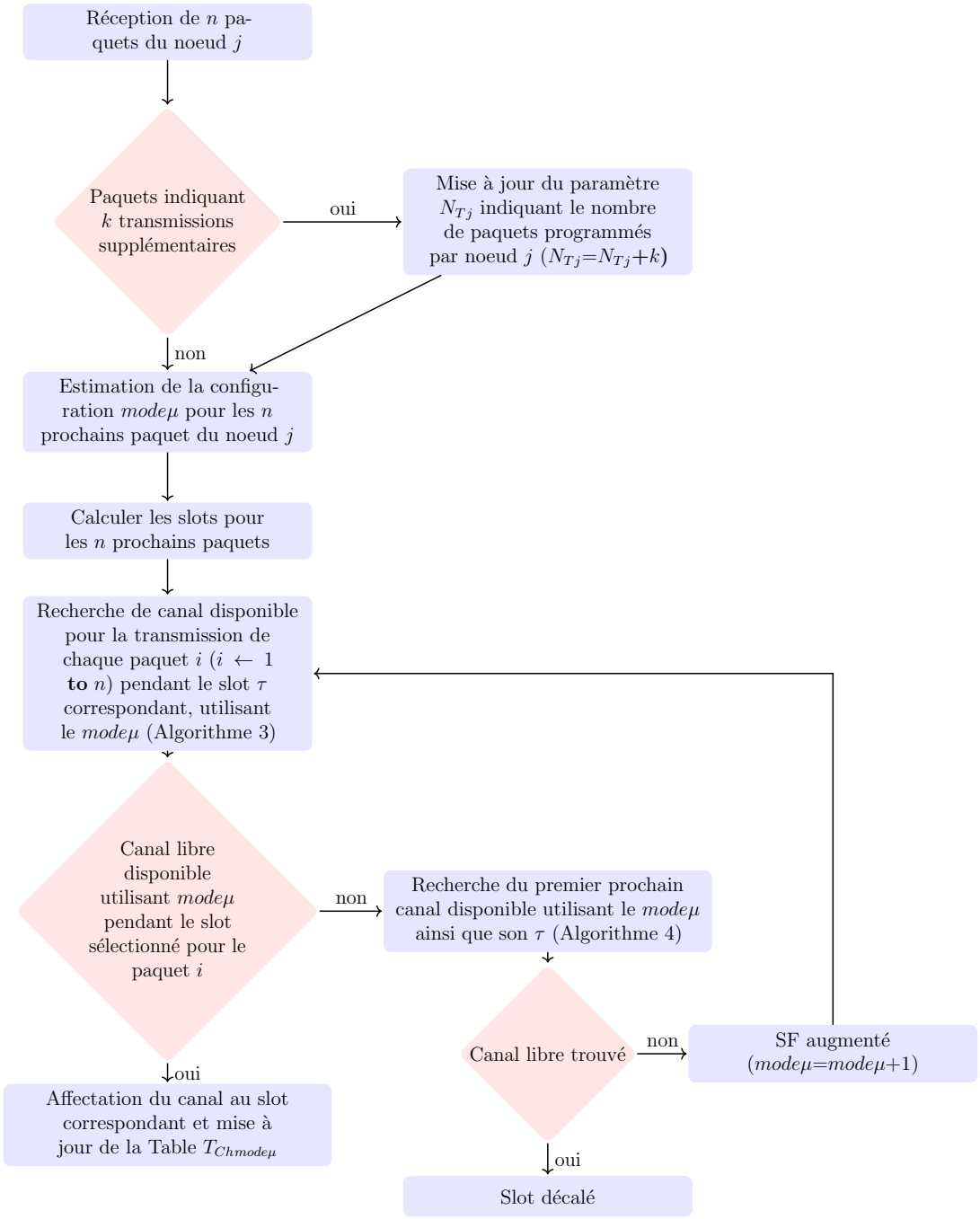


FIGURE IV.11 – Allocation de Slot/Canal par le serveur

3.2.2 Détection de données supplémentaires

Le noeud j est programmé pour envoyer une donnée chaque $P_{sending(j)}$. Afin de différencier les paquets réguliers des paquets supplémentaires, La Figure IV.12 montre que lors de la détection des données prêtes à être envoyées, le noeud j vérifie si le temps $t_{((i+1),j)start}$ du début de transmission $(i+1)$ suit le rythme régulier d'envoi de paquets réguliers $P_{sending(j)}$. Ceci en vérifiant si l'instant d'envoi du paquet $(i+1)$ est définit par : $t_{((i+1),j)start} = t_{(i,j)end} + P_{sending(j)}$.

Algorithme 4 : Recherche du premier prochain canal disponible utilisant le $mode\mu$ ainsi que son τ pour un noeud j

input : τ=[$t_{(i,j)start}$, $t_{(i,j)end}$] : slot calculé pour le paquet i du noeud j ; $mode\mu$: le mode sélectionné ; $T_{chmode\mu}$: table des slots et canaux pour le mode μ ; N_{ch} : nombre de canaux par mode ; $N_{\tau\mu}$: nombre de slot dans un canal utilisant $mode\mu$

output : C_l (Canal libre pendant le slot sélectionné)

```
1  for v ← (τ + 1) to N_τμ do
2      for w ← 1 to N_Ch do
3          X=T_Chmodeμ[v, w]
4          if X==0 then
5              C_l = w & τ=v ; mettre à jour T_Chmodeμ[v, w] (T_Chmodeμ[v, w]==1) ; exit
6          else if X≠ 0 then
7              w = w + 1
8      end
9  end
10 if C_l == 0 then
11     v = v + 1
12 else if C_l ≠ 0 then
13     Affecter le slot τ utilisant C_l au noeud j pour sa i^ème transmission ; exit
```

Si la vérification est "vrai", les noeuds devraient envoyer leurs données dans les slots déjà programmés par le serveur sans l'alerter sur la détection de données supplémentaires. Dans le cas contraire, les noeuds devraient d'une part attendre le prochain slot programmé à $t_{((i+1),j)start}$ pour envoyer leur donnée détectée dans l'intervalle $\tau = [t_{(i,j)end}, t_{((i+1),j)start})$ et d'une autre part informer le serveur des k données supplémentaires détectées à travers une nouvelle commande MAC appelée $IrregularTx$ au CID=$0x2B$.

3.3 Évaluation de performances

Cette section présente les résultats expérimentaux du schéma d'allocation dynamique SF-Slot-Canal. Nous évaluons dans cette section les performances de notre contribution et son efficacité dans le contexte d'un réseau de noeuds statiques et mobiles. Pour valider notre contribution, les performances sont évaluées en termes de PLR et de consommation énergétique.

Dans ce qui suit, nous décrivons le scénario de test, les détails du paramétrage expérimental, les métriques d'évaluation et les résultats.

3.3.1 Scénario de test

Pour démontrer l'efficacité de notre schéma d'allocation dynamique, nous avons déployé un total de 15 nœuds (trois waspmote SX1272 [49] [83], deux STM32 Discovery [84], dix Lopy4 LoRaWAN [90]) dans un étage du laboratoire. Sur les 15 nœuds, 5 sont des dispositifs mobiles et 10 sont statiques. Notre but est de se retrouver dans un cas où les noeuds doivent

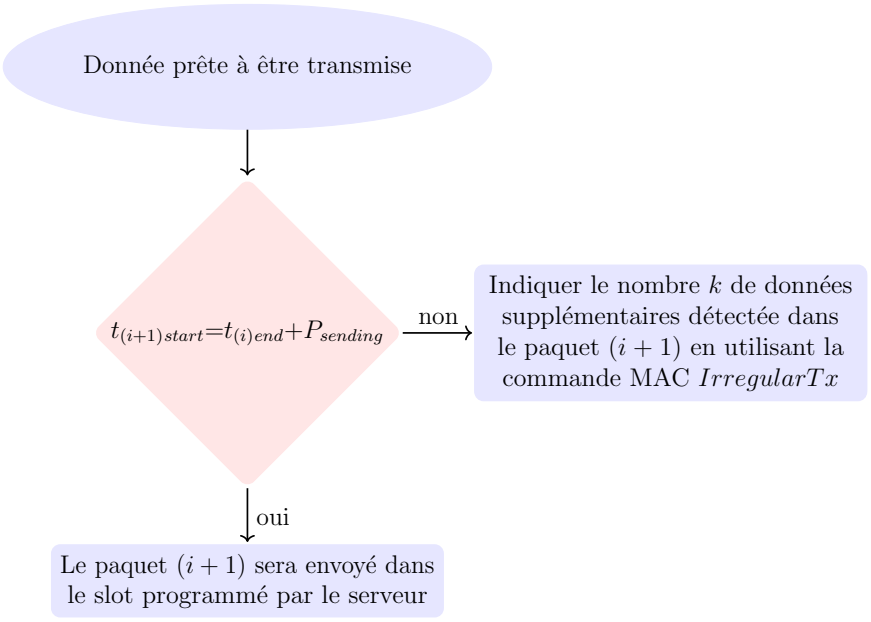


FIGURE IV.12 – Détection des données supplémentaires par le noeud

attendre la programmation d'un autre slot pour envoyer leurs transmissions ou changer de mode de configuration pour une raison d'insuffisance des canaux de transmission. Pour cela, nous avons autorisé l'utilisation de seulement 3 canaux pour la transmission des paquets dans la liste Ch afin que chaque mode de configuration puisse être utilisé au maximum par 3 transmissions simultanées.

Les noeuds $D1$-$D5$ sont des noeuds mobiles. Les noeuds $D6$-$D15$ sont fixes. Les besoins de $D1$-$D5$ en termes de ressources (portée) changent en fonction de leurs déplacements, d'où la nécessité d'une allocation dynamique de slots au fur et à mesure des déplacements des noeuds. De plus, certains noeuds feront face à des événements inattendus nécessitant la programmation de slots additionnels pour l'envoi des données supplémentaires (noeuds irréguliers). Nous supposons que les différents noeuds s'enregistrent auprès du serveur à travers l'envoi du message Join Request sur des intervalles de temps différents en utilisant $mode1$ afin de garantir la réception du message et d'éviter les collisions des messages de jointure. En recevant le message de jointure, et à partir de l'ID de l'application à la quelle le noeud est dédié, le serveur prend connaissance du type de noeud, nombre de paquets à transmettre, type d'application, fréquence d'envoi, etc). La Table IV.6 décrit les caractéristiques des différents noeuds enregistrés au niveau du serveur après la procédure de jointure. Dans le cas des noeuds fixes, le RSSI reçu lors du Join Request définira le mode de configuration utilisé (qui va être vérifié après chaque n paquets reçus). Cependant, les modes de configuration des noeuds mobiles sont variables selon leurs déplacements et seront attribués selon la stratégie VHMM-based E-ADR après la réception de chaque paquet ($n = 1$). Tous les paquets envoyés par les différents noeuds sont de taille de $20Bytes$ chacun.

Noeuds	type	Nombre de paquets		$P_{sending(i)}$	SF
		réguliers	irréguliers		
D_1, D_4	mobile, irrégulier	9	3	1pkt/5min	variable
D_2, D_5	mobile, irrégulier	13	2	1pkt/3min	variable
D_3	mobile	13	0	1pkt/4min	variable
D_6, D_7	fixe, irrégulier	9	3	1pkt/3min	SF11
D_8	fixe	6	0	1pkt/9min	SF11
D_9	fixe	8	0	1pkt/6min	SF11
D_{10}	fixe	10	0	1pkt/6min	SF10
D_{11}	fixe, irrégulier	8	4	1pkt/3min	SF10
D_{12}, D_{13}	fixe, irrégulier	11	2	1pkt/3min	SF9
D_{14}	fixe	8	0	1pkt/6min	SF9
D_{15}	fixe	5	0	1pkt/12min	SF9

Table IV.6. caractéristiques des noeuds

3.3.2 Évaluation du taux de collision

Dans nos expérimentations, l'utilisation d'E-ADR considère qu'un paquet ne peut être perdu qu'à cause d'une mauvaise allocation SF limitant sa portée par rapport à la gateway. De plus, vu la petite quantité de données que les noeuds envoient, la possibilité de pertes dues au dépassement de duty cycle est écartée. Nous nous concentrons alors sur le taux de collision ($T_{collision}$) dû à l'utilisation de la technique P-ALOHA. Les paquets qui entrent en collision ne sont pas forcément perdus, car ils vont être retransmis (Trafic Confirmé). Nous utilisons les configurations de la Table II.1, où la puissance $TP = 14dB$. Ceci dit, l'effet de capture [37] n'est pas applicable. Lorsque des noeuds entrent en collision (s'interfèrent), ils ne sont pas reçus et doivent être retransmis.

Parmi les différents tests effectués (12 tests), nous avons présenté dans les Figures IV.13-IV.14 un scénario afin de comparer les différentes combinaisons implémentées. Nous allons comparer les deux différentes combinaisons en termes de taux de collision ($T_{collision}$) du réseau LoRaWAN défini par le nombre de paquets entrant en collisions sur le nombre total des paquets envoyés par tous les noeuds. Le but est de discuter l'apport de l'approche déterministe (SF-Slot-Canal) par rapport à la combinaison "SF ALOHA" et de montrer l'importance de la sélection des slots selon la disponibilité des canaux en écartant le comportement ≪ pseudo aléatoire ≫.

La Figure IV.13 présente la combinaison SF ALOHA. Nous rappelons que dans ce cas, le numéro de canal à utiliser est sélectionné d'une manière pseudo aléatoirement [21] entre 1 et 3 (3 canaux seulement) et les paquets sont envoyés directement suite à leurs détection.

La Figure IV.14 présente la solution SF-Slot-Canal. Les slots de transmission sont attribués selon la disponibilité des canaux dans les listes $T_{Chmode\mu}$.

Dans la Figure IV.13, les paquets envoyés par les noeuds j utilisant le même mode $mode\mu$ (de même couleur sur la Figure IV.13), se trouvant sur le même instant sur le même canal (numéroté de 1 à 3), s'interfèrent et ne sont pas reçus puisqu'ils utilisent la même puissance $TP = 14dBm$ (Pas d'effet de capture) [37], ce qui augmente le taux de collision ($T_{collision}$). Prenons par exemple les paquets envoyés à t_9 par D_2, D_6, D_7, et D_8, utilisant $mode3$ (Figure

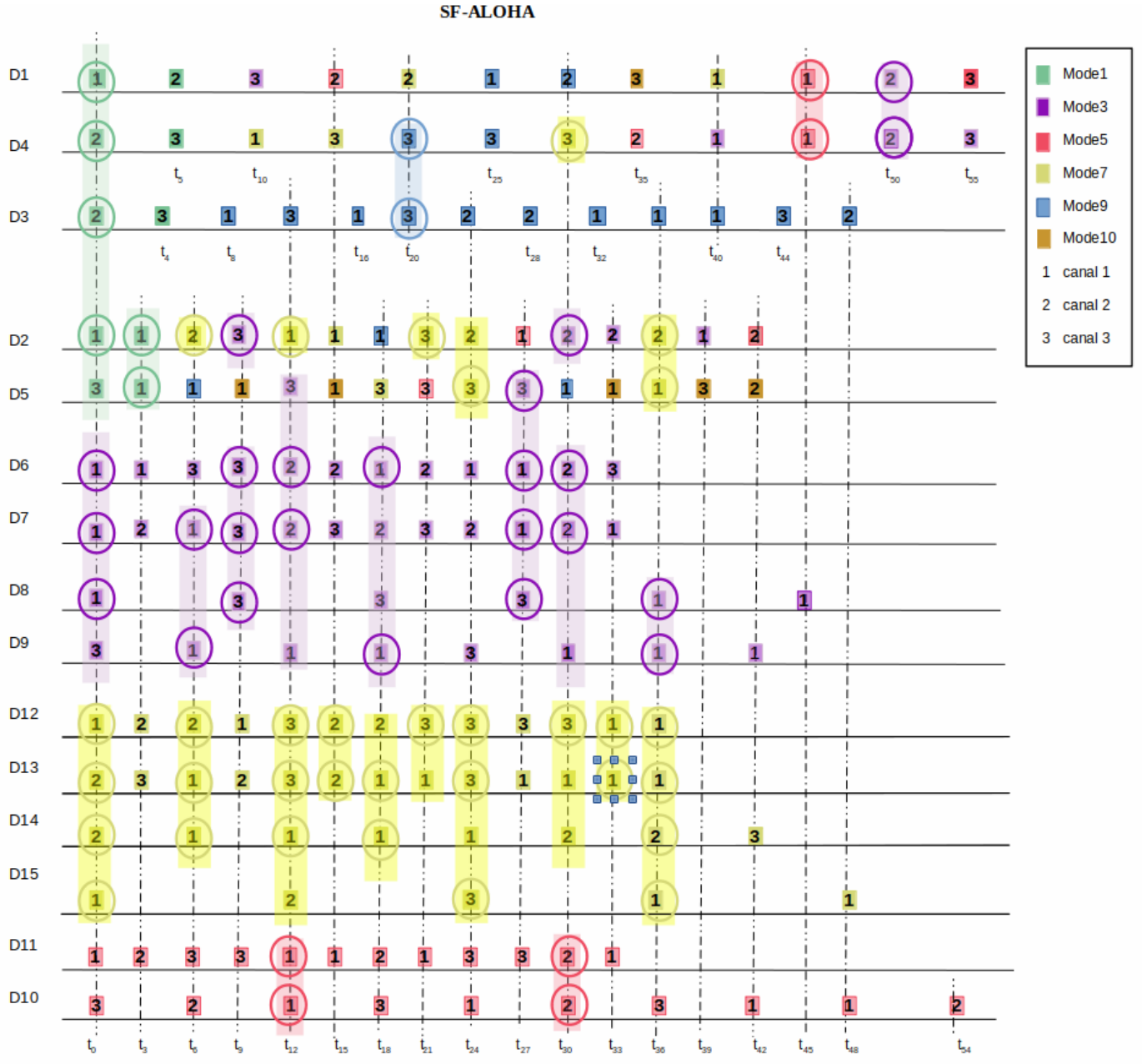


FIGURE IV.13 – SF-ALOHA

IV.13). La sélection pseudo-aléatoire a fait que les 4 paquets sont envoyés sur le *canal*3 provoquant des interférences et induisant leurs retransmissions.

Dans la Figure IV.14, nous constatons que l'approche SF-Slot-Canal a permis à l'instant t_9, par exemple, de transmettre 3 paquets sur différents canaux et de programmer la $4^{ème}$ transmission sur un autre slot sur le 1^{er} prochain canal disponible, qui est le *numéro*1.

La Table IV.7 présente le taux moyen de collision $T_{collision}$ des 12 tests effectués utilisant les différentes combinaisons. Les $T_{collision}$ obtenus à partir du scénario présenté dans les Figures IV.13- IV.14 pour les combinaisons "SF-ALOHA", et "SF-Slot-Canal" sont respectivement :

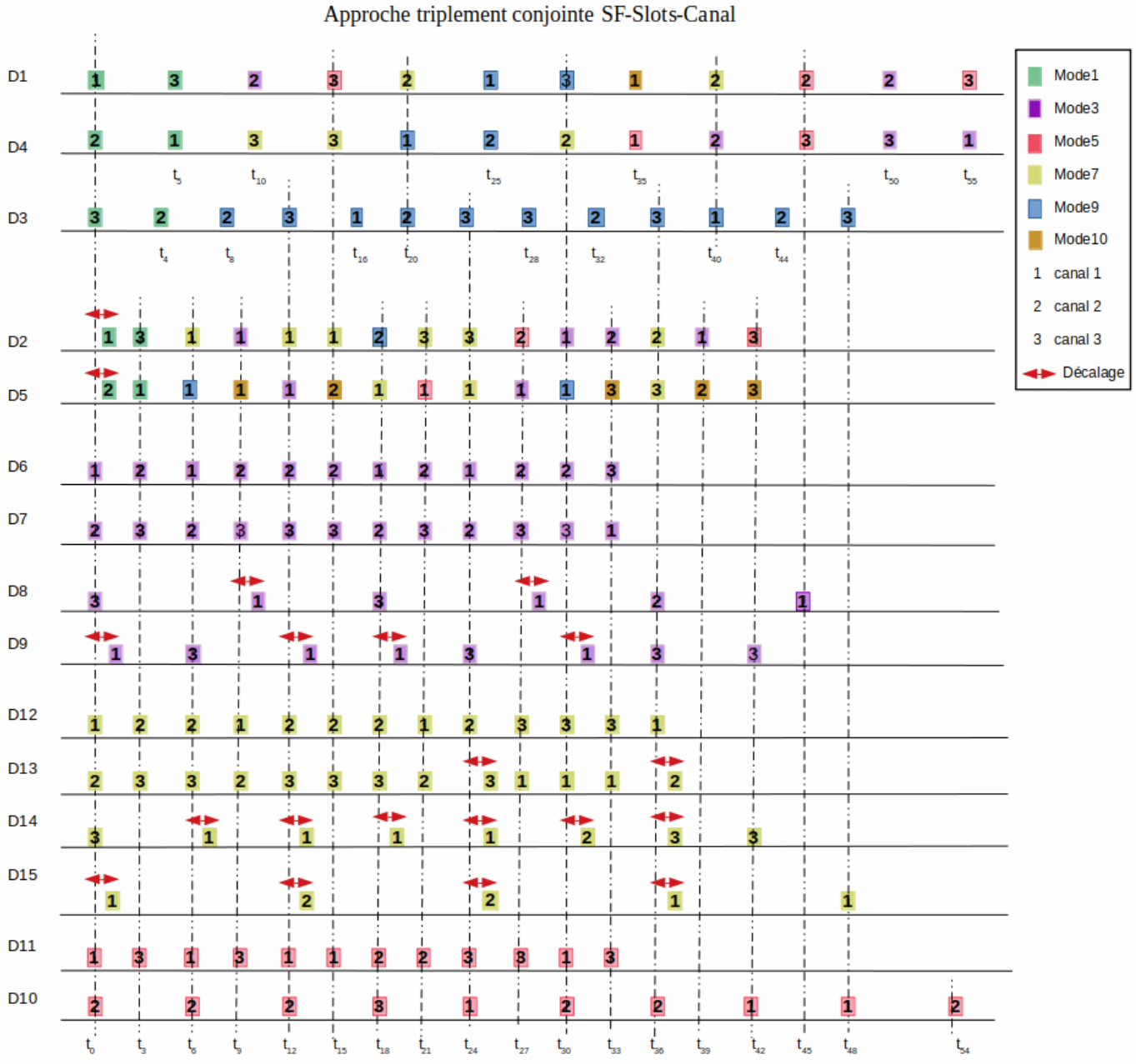


FIGURE IV.14 – SF-Slot-Canal

40.89%, et 0%. La triple conjonction SF-Slot-Canal a permis d'éliminer toute probabilité de collision et de bénéficier de la réception de tous les paquets dès la première tentative de transmission écartant toute retransmission de paquet en contrepartie d'un décalage d'envoi (flèche rouge). L'impact de ce décalage sera analysé plus loin.

Table IV.7. $T_{collision}$ moyen du réseau LoRaWAN expérimenté

	SF-ALOHA	SF-Slot-Canal
$T_{collision}(\%)$	40.89%	0%

3.3.3 Évaluation de l'équité dans l'occupation des canaux "Load Balancing"

Le Load balancing dans les canaux est un critère important. Il est conçu pour assurer une haute disponibilité et une grande fiabilité et scalabilité du réseau. Il peut améliorer le taux de réception des paquets et augmenter les performances du réseau en utilisant pleinement les ressources disponibles. Le but est d'équilibrer le trafic sur les différents canaux afin de réduire les collisions et de prolonger la durée de vie et la disponibilité des canaux face à une restriction de DC des gateways. Pour évaluer ce critère, nous présentons dans la Figure IV.15 le taux d'occupation des canaux obtenus à partir des Figures IV.13- IV.14.

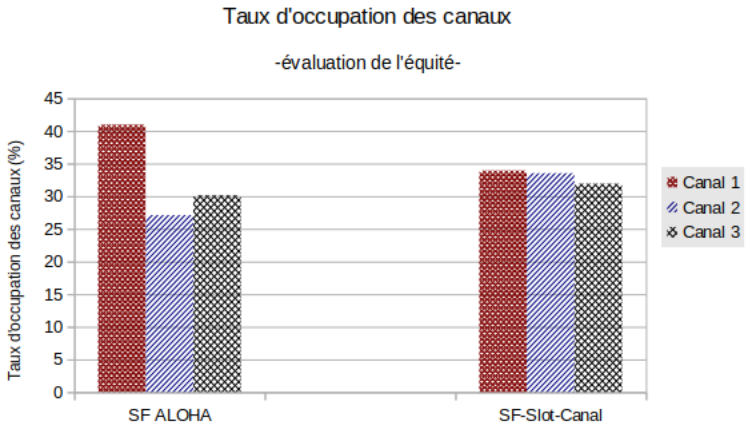


FIGURE IV.15 – Évaluation du taux d'occupation des canaux

Nous observons dans cette figure que le taux d'utilisation des canaux est variable avec une différence de marge plus importante dans le cas des combinaison "SF ALOHA" par rapport à "SF-Slot-Canal" à cause de la sélection pseudo-aléatoire des canaux. Cette distribution non équilibrée augmente la probabilité de collisions dans les canaux fortement chargés induisant une augmentation du PLR. De plus, l'utilisation d'un canal chargé risque d'épuiser le DC autorisé par rapport aux autres canaux moins chargés. La combinaison SF-Slot-Canal permet d'avoir un certain niveau d'équité en balançant le trafic équitablement sur tous les canaux disponibles et résulte en un taux d'occupation moyen de 33.12% avec une variabilité de $\approx \pm 1.5\%$.

3.3.4 Évaluation du ToA et de la consommation énergétique

Le temps de transmission (ToA) ainsi que la consommation énergétique sont des critères importants dans l'optimisation de la QoS du protocole LoRaWAN. En effet, l'idée est de réduire le $T_{collision}$, mais tout en assurant une consommation énergétique optimale. Les paquets entrants en collision sont tous retransmis jusqu'à ce qu'ils soient correctement reçus afin de réduire le PLR. Ces retransmissions résultent en une augmentation du ToA et par conséquent une consommation énergétique élevée.

Dans cette partie, nous évaluons le ToA consommé par tous les noeuds lors de l'utilisation des différentes combinaisons afin de valider l'efficacité de notre approche "SF-Slot-Canal". Les Tables IV.8 et IV.9 présentent respectivement le ToA consommé par tous les noeuds dans

les 2 cas. En envoyant une faible quantité de données, nous supposons que le DC n'a pas été dépassé par les noeuds dans le cas de SF-ALOHA même en ayant recours aux retransmissions.

Table IV.8. ToA consommé par tous les noeuds en utilisant les différentes combinaisons (ms)

	SF-ALOHA	SF-Slot-Canal
ToA (ms)	37749.89	25250.49

Table IV.9. Consommation énergétique moyenne par paquet reçu (J)

	SF-ALOHA	SF-Slot-Canal
$\acute{E}nergie_{moyenne}$ (J)	0.212	0.147

En faisant face à un taux de collisions significatif (Table IV.7), la combinaison « SF-ALOHA » se voit retransmettre tous les paquets interférés. Bien que ces retransmissions offrent une chance aux paquets d'être correctement reçus, ces dernières augmentent significativement le ToA ($37749.89ms$) et par conséquent la consommation énergétique moyenne par paquet ($0.212J$). D'autre part, nous observons que le ToA consommé en utilisant notre approche SF-Slot-Canal est diminué avec l'atténuation de la probabilité de collision. En effet, l'affectation déterministe des slots permet d'écarter les retransmissions et donc de bénéficier d'une faible consommation du ToA ($25250.49ms$) et par conséquent une meilleure efficacité énergétique ($0.147J$ par paquet).

3.3.5 Évaluation du délai de transmission

Le délai de transmission peut être impacté par différents facteurs, par exemple, dans notre cas la programmation d'un slot retardé. Mais, l'un des facteurs le plus important reste le nombre de retransmissions. Dans cette partie, nous allons évaluer le délai de transmission moyen par paquet pour les deux protocoles testés.

La Table IV.10 présente le délai de transmission moyen d'un paquet utilisant les différentes combinaisons : « SF-ALOHA », et « SF-Slot-Canal ».

Table IV.10. Délai de transmission moyen par paquet (TD) (ms)

	SF-ALOHA	SF-Slot-Canal
TD (ms)	1073.63	952.11

Bien que l'approche « SF-Slot-Canal » décale des slots de transmission lorsqu'aucun canal n'est libre, augmentant le délai de transmission, elle permet d'atténuer toutes les collisions et par conséquent aucune retransmission n'aura lieu. Cette augmentation reste négligeable devant l'augmentation causée par les retransmissions dans le cas de SF-ALOHA. En résumé, en plus du taux de collision significatif obtenu par l'utilisation de « SF-ALOHA », le délai moyen de transmission de paquet est aussi augmenté à cause des retransmissions des paquets interférés, alors que notre approche SF-slot-canal permet de réduire le délai de transmission en évitant les collisions (donc les retransmissions).

4 – Conclusion

Après avoir présenté les inconvénients de DC fixe, nous avons d'abord présenté un mécanisme de DC dynamique partagé entre les noeuds dans le réseau afin d'améliorer la QoS de LoRaWAN en augmentant le taux de réception de paquets et le nombre de noeuds ayant eu la chance de finir toutes leurs transmissions et évaluer ses performances par rapport au DC fixe expérimentalement. Comme amélioration de l'utilisation du DC dynamique, Après avoir présenté la limitation liée au DC fixe, nous avons proposé de combiner les deux propositions « E-ADR - DC dynamique » afin de garantir une meilleure QoS du réseau LoRaWAN. Les résultats obtenus à partir de nos expérimentations montrent d'amples améliorations grâce à cette combinaison ainsi qu'un meilleur compromis « PLR - consommation énergétique ».

Ensuite, nous avons proposé dans une deuxième partie de ce chapitre, une approche triplement conjointe SF-Slot-Canal, qui donne naissance à un protocole LoRaWAN amélioré « E-LoRaWAN ». Cette approche permet l'allocation dynamique et déterministe des paramètres de configuration sur des canaux de fréquence et des slots de transmission efficaces. Le but est d'améliorer le taux d'occupation des ressources et les performances du réseau. Nous avons évalué notre solution en utilisant notre plate-forme expérimentale LoRaWAN. Les résultats de cette évaluation ont montré que la combinaison SF-Slot-Canal permet d'améliorer la QoS du réseau en éliminant toute probabilité de collision et en garantissant un bon compromis entre l'efficacité énergétique et le taux de réception des paquets transmis tout en équilibrant l'utilisation des canaux réduisant ainsi le ToA. Ceci prouve l'efficacité de notre E-LoRaWAN.

Chapitre V

Étude de la vulnérabilité de E-LoRaWAN

Dans ce chapitre, nous examinons la vulnérabilité du protocole E-LoRaWAN construit par la combinaison des solutions proposées dans les chapitres précédents. Nous nous concentrons plus précisément sur l'influence des attaques « Bit-flipping » et « ID-Spoofing » sur le réseau E-LoRaWAN en analysant les risques de ces attaques sur la performance du réseau. Pour cela, nous étudions un scénario impacté par ces deux attaques auquel nous proposons une solution de sécurisation par la suite. Cette solution est implémentée et évaluée sur les modules SX1272 de Waspmote de libelium et STM32 de STMicroelectronics.

1 – Introduction

Le protocole LoRaWAN de base étant relativement récent, son niveau de sécurité n'est pas très développé, ce qui le rend vulnérable à plusieurs attaques telles que les attaques de relecture "Replay", les attaques de retournement de bits "Bit-flipping", etc. Par conséquent, le développement de solutions de sécurité légères est essentiel pour les petits périphériques LoRaWAN qui sont limités en termes d'énergie et de capacité de traitement.

Bien que LoRaWAN fournisse des mécanismes de sécurité, tels que le cryptage et la signature, son niveau de sécurité reste limité et le risque encouru n'est pas négligeable. En effet, LoRaWAN génère dynamiquement deux clés AES de 128 bits lors de l'activation du noeud qui seront utilisées respectivement pour vérifier l'intégrité et pour chiffrer les données jusqu'au serveur d'application. Cependant, ces clés peuvent être sujets à des attaques. Les données cryptées également peuvent être modifiées, etc. En outre, notre protocole E-LoRaWAN posera les mêmes risques que LoRaWAN. Les mécanismes E-ADR et Duty Cycle Dynamique utilisés peuvent être attaqués, ce qui nuit à la QoS du réseau. Nous présentons dans la suite quelques fonctions de sécurité prévues dans LoRaWAN ainsi que des exemples de vulnérabilités et d'attaques possibles. Nous nous intéressons à deux exemples d'attaques qui peuvent nuire au fonctionnement de E-LoRaWAN et les solutions envisagées.

2 – Les caractéristiques de sécurité LoRaWAN

Bien que la sécurité n'est pas très développée dans le protocole LoRaWAN, ce dernier a défini différentes techniques pour renforcer son fonctionnement [21]. Dans ce qui suit, nous

listons ces différents aspects et nous discuterons l'avantage et l'inconvénient du niveau de sécurité implémentée par LoRaWAN.

2.1 Méthode OTAA

Nous rappelons que la procédure OTAA (présentée dans la section 3.4.5) consiste en un échange d'un « Join Request » et d'un « Join Accept » entre le noeud et le serveur pour assurer l'enregistrement des noeuds dans le réseau LoRaWAN. Dans la demande de jointure (Join Request), chaque noeud devrait connaître son unique identifiant ($DevEUI$), l'identifiant unique de son application ($AppEUI$), et devrait générer une séquence aléatoire unique appelée ($DevNonce$). Cette demande de jointure n'est pas cryptée. Alors que la réponse de jointure (Join Accept) est cryptée par une clé $AppKey$ échangée entre le noeud et le serveur au préalable [34]. Cette réponse contient un identifiant unique ($AppNonce$) qui servira au noeud pour générer les clés de session (Nwk_SKey, App_SKey) comme c'est fait au niveau du serveur, qui serviront pour le chiffrement et la vérification des paquets de données échangés entre le noeud et le serveur. Par conséquent ces clés de session ne sont pas échangées entre le noeud et le serveur, et elles sont re-générées à chaque enregistrement ou ré-initialisation. La procédure de génération des clés est présentée dans le Chapitre 1 (Section 3.4.7).

OTAA assure un certain niveau de sécurité. Premièrement, il utilise des paramètres uniques par noeud tels que, $AppKey$, $DevEUI$, $AppEUI$, $AppNonce$ et $DevNonce$. Dans ce cas, compromettre un seul noeud ne signifie pas compromettre l'ensemble du réseau. Deuxièmement, les valeurs $DevNonce$ sont enregistrées afin d'éviter certaines attaques. Chaque fois qu'une demande de nouvelle connexion est reçue, le serveur doit vérifier que le $DevNonce$ n'a pas été utilisé auparavant. Sinon, cette demande est rejetée pour éviter une attaque de "Replay" ayant pour objectif de déconnecter le noeud en question. En outre, au lieu de transmettre directement des clés comme dans les réseaux traditionnels, dans LoRaWAN, c'est les $DevNonce$ qui sont transmis et ce n'est que lorsque les clés $AppKey$, $DevNonce$ et $AppNonce$ (qui est cryptée) sont toutes obtenues que le tiers peut en déduire des clés de session. Dans ce cas, la difficulté d'obtenir les clés et de compromettre le réseau est plus importante.

2.2 Cryptage CounTeR (CTR)

Lors des communications, la charge utile du paquet est d'abord cryptée en se basant sur le cryptage AES-128 en mode CounTeR (CTR). Si la charge utile ne contient que des commandes MAC, la clé $NwkSKey$ est utilisée pour le cryptage. Sinon, la clé $AppSKey$ est utilisée. Le processus de cryptage est présenté dans le Chapitre 1 (Section 3.4.5). Dans le protocole LoRaWAN, une petite modification a été apportée au mode CTR. Contrairement au mode CTR de base, où le compteur de bloc contient un paramètre $Nonce$, LoRaWAN utilise les compteurs $FCntUp$ et $FCntDown$ qui correspondent respectivement au nombre de trames envoyées par le noeud et reçus de la part du serveur. Toutefois, lorsque ces deux compteurs $FCntUp$ et $FCntDown$ sont égaux, le mode est identique au mode CTR de base. La Figure V.1 présente la différence entre CTR et CTR-LoRaWAN.

L'inconvénient de CTR et CTR-LoRaWAN est que la taille du message en clair es t identique à la taille du message chiffré correspondant sans changement de format ni de

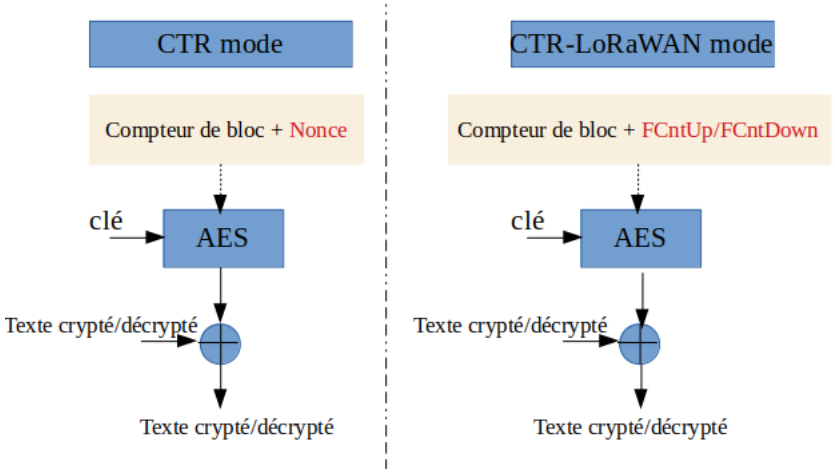


FIGURE V.1 – Mode CTR Vs. Mode LoRaWAN-CTR

remaniement des positions des bits. Ceci augmente la vulnérabilité du mode CTR à plusieurs attaques telles que "Man in the middle" et "Bit-flipping", et donc son utilisation n'est pas du tout recommandée dans les protocoles cryptographiques [110, 111].

2.3 Authentification par MIC

Le code d'intégrité des messages (MIC) est utilisé dans LoRaWAN pour assurer un contrôle d'intégrité. La charge utile MAC des messages, envoyés et/ou reçus, est signée pour empêcher la manipulation du texte chiffré. Le noeud calcule le MIC avant la transmission du paquet. Ce dernier sera recalculé par le serveur et les deux MIC seront comparés afin de vérifier l'intégrité du paquet transmis. Si jamais les deux MICs sont différents, le paquet est rejeté.

Les messages dans les réseaux LoRaWAN doivent être signés et cryptés. Deux types de paquets sont examinés dans cette partie pour montrer les différences de signature et de cryptage. Ces deux types de paquets sont le paquet d'acceptation de jointure (Join Accept) et le paquet de données. Pour les paquets Join Accept, la méthode "signer puis crypter" est utilisée. Pour les autres paquets de données, on utilise plutôt "crypter puis signer". La principale différence entre ces deux méthodes est de savoir si l'intégrité du texte chiffré est assurée. La méthode "signer puis crypter" ne garantit pas l'intégrité du texte chiffré, car ce n'est qu'après le décryptage que nous saurons si l'intégrité est maintenue ou non. Sans l'intégrité du paquet, il est possible que les attaquants puissent créer des cryptogrammes différents qui se décryptent correctement : "Crypter puis signer" assure l'intégrité du cryptogramme. Cependant, il existe des attaques qui peuvent impacter le texte crypté si sa taille est identique au texte en clair.

Les caractéristiques de sécurité dans LoRaWAN offrent plusieurs avantages, mais restent insuffisantes et représentent une forte vulnérabilité comme toutes les technologies existantes. Dans ce qui suit, nous présentons les différentes attaques contre le protocole LoRaWAN.

3 – Les attaques contre les réseaux LoRaWAN pour une jointure OTAA

Il existe plusieurs attaques de sécurité pouvant compromettre un réseau LoRaWAN comme les attaques : d'écoute (eavesdropping), de relecture (replay), de falsification des messages, de déni de service (DoS), etc. Nous présentons dans cette partie quelques attaques et leurs influences sur le système LoRaWAN.

3.1 Attaque d'écoute "Eavesdropping"

L'attaque est conçue pour compromettre la méthode de cryptage de LoRaWAN. En interceptant le trafic sans fil entre la gateway et le noeud, l'attaquant peut utiliser la relation correspondante entre deux messages ayant la même valeur de compteur pour déchiffrer le texte chiffré.

Pour que l'attaque puisse avoir lieu, il faudrait que l'attaquant arrive à intercepter deux paquets différents utilisant la même valeur du compteur. Pour se faire dans OTAA, il faudrait attendre la ré-initialisation du compteur pour faire une deuxième capture d'un message avec le même compteur de la première capture.

Lors d'une ré-initialisation, la valeur du compteur est remise à zéro conformément aux spécifications [21] tant que la clé reste en place, cela signifie que le chiffrement par blocs recréera exactement les mêmes clés App_SKey et Nwk_SKey. Après la ré-initialisation du compteur, toutes les valeurs vont être réutilisées avec les mêmes clés de session, et donc la valeur du compteur interceptée au début par le noeud va être réutilisée et capturée par l'attaquant un seconde fois. Ayant les captures (C1 et C2) utilisant la même valeur du compteur et la même clé, la cryptographie pourrait être compromise comme suit : Étant donné que le cryptage par CTR est effectué en XORant le texte brut P avec les clés K, il est possible de XORer deux messages cryptés par capture (C1 et C2) avec le même compteur et d'obtenir le texte en clair (P1 et P2) correspondant à (C1 et C2) (Voir Eq V.1) [112].

$$C1 \oplus C2 = (P1 \oplus K) \oplus (P2 \oplus K) = (P1 \oplus P2) \oplus (K \oplus K) = P1 \oplus P2 \tag{V.1}$$

Plus il y a de ré-initialisations, plus il y a de chances de récupérer les messages. Pour prévenir cette attaque, l'utilisation d'un mode de cryptage plus complexe est préférable.

3.2 Les attaques par brouillage "Jamming"

Les attaques par brouillage (Jamming) sont extrêmement difficiles à éviter, et c'est l'un des problèmes les plus difficiles auxquels sont confrontées les technologies sans fil, particulièrement dans le domaine de l'IoT. L'attaque par brouillage consiste à transmettre un signal radio à la même fréquence (porteuse) qu'une transmission radio en cours pour la perturber. Dans ce contexte, il est possible de brouiller la réception à un nœud ou à une gateway menant à un déni de service (DoS). Si un attaquant place un brouilleur à côté d'une gateway, il est très facile de détecter cette attaque, car les administrateurs du serveur peuvent voir qu'aucun appareil ne transmet vers cette gateway. Cependant, les attaques de brouillage sélectif (Selective

Jamming) sont plus difficiles à détecter et très dommageables pour les communications sans fil, car il n'est pas anodin de les éviter. C'est pourquoi, dans LoRaWAN, il est possible de brouiller la réception des signaux au niveau d'une passerelle ou d'un nœud en utilisant le brouillage simple. Ainsi, cela pourrait ouvrir la voie à des attaques avancées, comme l'attaque par relecture (Replay Attack), qui repose en partie sur la possibilité de réaliser un brouillage sélectif.

Les attaques de brouillage peuvent être évitées en détectant les noeuds ayant un comportement anormal. Cela peut être fait pendant l'attaque, puisque tous les noeuds de communication malveillants seront détectés et retirés du réseau. En outre, pour préserver la disponibilité des noeuds, l'administrateur réseau peut commuter la transmission sur un autre canal de transmission sur une autre bande de fréquence.

3.3 Attaque par usurpation d'identité "Spoofing"

Une attaque Spoofing est une attaque utilisée par un noeud malveillant pour se déguiser et se faire passer pour un autre noeud ou usurper les informations des autres noeuds. Il existe plusieurs types d'attaques Spoofing : ACK-Spoofing (usurpation des ACKs), ID-Spoofing (usurpation d'identité, ça peut être n'importe quelle information unique à un noeud), IP-Spoofing (usurpation d'adresse IP), GPS-Spoofing (usurpation des signaux GPS), etc. En usurpant ces informations, un attaquant peut bénéficier de l'accessibilité des différents services gratuitement.

Comme dans le cas de LoRaWAN, la demande de jointure « Join Request » n'est pas cryptée, alors ce protocole reste vulnérable à une usurpation des données comme l'identifiant d'un noeud ($DevEUI$), l'identifiant de l'application ($AppEUI$) ainsi que les informations concernant le duty cycle dynamique ou l'E-ADR que nous avons proposées dans les chapitres précédents. Dans notre cas, l'attaque ID-Spoofing peut être faite dans le but de se faire passer pour un demandeur de priorité élevé afin de bénéficier d'un temps additionnel sans en avoir besoin et de priver le vrai demandeur de finir sa transmission tout en dégradant le PLR ainsi que la QoS du réseau E-LoRaWAN intégrant le mécanisme de duty cycle dynamique proposé dans le Chapitre 3. Autre possibilité est qu'il suffit de modifier la valeur de RSSI d'un paquet pour fausser le résultat et donc donner une mauvaise configuration au noeud correspondant à cette transmission.

3.4 Attaque par retournement de bits "Bit-flipping"

L'attaque Bit-flipping est une attaque où un noeud malicieux peut modifier un champ spécifique dans le texte chiffré sans avoir à le décrypter [113]. L'attaque par bit-flipping est possible dans les modes de cryptage où un texte en clair a le même ordre de bits que le texte chiffré [114]. L'attaque se fait sur le message crypté en modifiant un octet du texte chiffré dont la disposition des octets est connue. Le mode de cryptage CTR (Counter) utilisé dans LoRaWAN est vulnérable à cette attaque [115, 116], parce qu'il effectue une simple opération « Xor » entre la clé et la donnée qui n'altère pas l'ordre des bits du texte en clair (Eq (I.13), Eq (I.14), Eq (I.15)). Le texte chiffré non mélangé permet à l'attaque Bit-flipping d'influencer les messages envoyés.

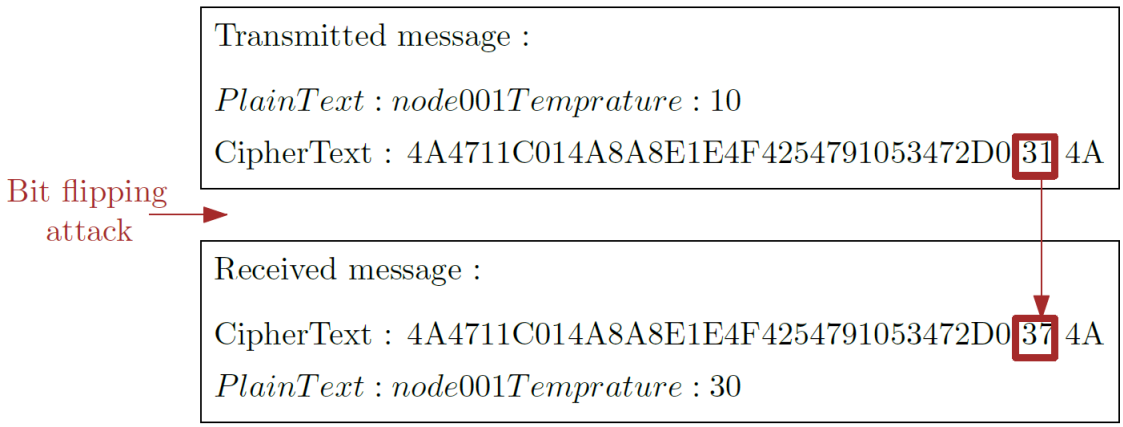


FIGURE V.2 – Attaque Bit-flipping

La Figure V.2 illustre l'attaque bit-flipping sur le texte chiffré en utilisant le mode CTR. Dans cette simulation, nous supposons que le format d'un paquet contenant une information sur la température est connu. Donc l'agresseur connaît l'emplacement des données du capteur de température. L'attaquant intercepte le texte chiffré et modifie 1 octet de la valeur des données du capteur dans le texte chiffré. En tant que résultat, le serveur du réseau LoRaWAN décrypte le message chiffré et obtient la mauvaise valeur des données des capteurs. Le mode CTR utilisé dans LoRaWAN est vulnérable à l'attaque Bit-flipping, car la position des octets du message n'est pas remaniée.

Cette attaque est souvent produite entre le noeud et le serveur réseau. L'objectif de cette attaque est de prouver que l'intégrité n'est pas protégée. Si l'attaquant a la possibilité de compromettre la transmission, il n'y a aucun moyen de l'empêcher avec les mécanismes de sécurité implémentés actuellement dans LoRaWAN consistant en un simple mode CTR.

Pour pouvoir effectuer cette attaque, l'attaquant doit connaître le format de la charge utile et le type du message envoyé par le noeud. Ces informations sont divulguées dans le Join Request envoyé par les noeuds, étant donné qu'il n'est ni sécurisé, ni crypté. De plus, l'attaquant, connaissant l'ID de l'application à laquelle le noeud est dédié (à partir du Join Request), ou bien en étant déjà informé du format du paquet dans le cas où le noeuds attaquant était un noeud saint lors de son enregistrement dans le réseau, il pourra facilement faire son attaque et dégrader la QoS du réseau.

Dans notre cas, cette attaque peut s'introduire dans E-LoRaWAN en modifiant l'information contenue dans le Join Request concernant le temps supplémentaire demandé par les noeuds (Duty Cycle dynamique) ou le RSSI calculé par la gateway et ajouté au paquet pour l'allocation de la configuration (E-ADR), l'ACK servant au changement du SF, par exemple. Le but est de forcer le serveur à rejeter le paquet envoyé par le noeud saint modifié par le noeud attaquant suite à la différence dans le calcul MIC. Le serveur en calculant le MIC du paquet attaqué, se rend compte que ce dernier n'est pas le même calculé par le noeud émetteur et finit par le rejeter. Ce rejet augmente le taux de perte (PLR) et pourrait donner plus de chance à un noeud attaquant, dans le cas où il est enregistré dans le réseau, de profiter d'un temps additionnel puisque le rejet du Join Request d'un noeud ayant besoin d'un temps supplémentaire signifie qu'un demandeur a été enlevé de la liste des demandeurs. Dans la

suite de ce chapitre, nous nous intéressons à l'analyse de cette attaque et à la proposition d'une solution.

4 – Les contre-mesures proposées pour le protocole LoRaWAN

Plusieurs travaux se sont intéressés à l'évaluation des différentes attaques, menaces et vulnérabilités du protocole LoRaWAN.

Les auteurs dans [117] ont discuté plusieurs vulnérabilités de LoRaWAN, telles que l'attaque Bit-flipping, ainsi que son impact sur la sécurité des serveurs LoRaWAN. En outre, [118] a présenté plusieurs solutions contre les attaques et les vulnérabilités de LoRaWAN et a défini comment la falsification de paquets peut être atténuée en utilisant le mode AES-CMAC pour l'authentification et la génération des clés au même temps avec une clé secrète de 128 bits.

Dans [119], une contre-mesure contre les attaques eavesdropping a été proposée. Celle-ci consiste à remplacer le numéro du compteur dans le message par un nonce de sorte que le message avec la même clé de session ne sera pas valide dans une autre session, mais les collisions de messages avec le même numéro peuvent se produire. La deuxième contre-mesure fournie consiste en un changement périodique de la clé de session.

Dans [112], la contre-mesure d'usurpation d'identité consiste à ajouter une somme de contrôle cryptographique de l'ensemble du paquet dans le message cible (ACK, par exemple). Cela permettra au nœud de vérifier si l'ACK reçu appartient au message envoyé et il peut également savoir si le message a été modifié. De cette façon, le message ACK est uniquement lié à un message et ne peut pas être utilisé pour accuser réception d'autres messages. Cependant, aucune solution n'a été proposée pour le cas des paquets de demande de jointure non cryptés.

Miller [115] donne un bref aperçu de la sécurité de LoRaWAN et explique comment configurer les dispositifs de sécurité dans le protocole pour la mise en place d'un réseau LoRaWAN. Il décrit l'emplacement du matériel clé dans une installation LoRaWAN, et les alertes qui présentent des failles dans la gestion des clés pouvant compromettre un back-end. Le travail n'analyse pas le protocole et n'évalue pas la sécurité des échanges de messages.

Un problème notoire en matière de sécurité des protocoles est l'utilisation insuffisante du caractère aléatoire ou nonce ("numéro utilisé une fois"). Les auteurs dans [120] et [121] analysent les menaces à la sécurité dans LoRaWAN en se concentrant sur la génération de DevNonce, qui est utilisé dans la procédure de jointure. Ils étudient le caractère aléatoire de DevNonce et fournissent des méthodes de génération alternatives. Les auteurs dans [122] se concentrent également sur les vulnérabilités de la procédure de jointure dans LoRaWAN.

Les auteurs dans [123] examinent la question des fuites de données personnelles dans le cas où des utilisateurs multiples tentent d'accéder au même serveur d'application. Il compare des algorithmes et des modes de cryptage en se basant sur le temps de calcul et les ressources utilisées. [124] compare les protocoles de gestion de clés existants pour l'IoT, et proposent d'ajouter un noeud proxy pour renforcer le mécanisme de sécurité de LoRaWAN.

[125] souligne les problèmes de la sécurité au niveau de la couche physique et développent des attaques pratiques autour du brouillage sélectif (Jamming). Des suggestions basées sur les caractéristique LoRaWAN sont données pour atténuer ces attaques. Parmi ces suggestions, l'augmentation du nombre de canaux obligeant l'attaquant à écouter tous ces canaux, ce qui rend l'attaque complexe. La seconde suggestion est de passer à un SF bas pour battre le

temps de réaction du brouilleur, etc.

Dans la suite, nous nous intéressons à des scénarios portant sur deux attaques : « Bit-flipping » et « ID-Spoofing », impactant sur le protocole E-LoRaWAN proposé dans les chapitres précédents, intégrant des champs contenant des informations importantes pouvant être usurpées. Dans ce qui suit, nous allons présenter les scénarios d'attaques.

5 – Étude d'exemples d'attaques du réseau E-LoRaWAN

Dans cette partie, nous nous intéressons à un scénario de 5 noeuds transmettant leurs données pendant un cycle d'une heure, qui est reproduit trois fois. D_1, D_2, D_3 et D_4 envoient respectivement 36, 44, 21 et 26 paquets de 50 Bytes chacun et sont demandeurs de respectivement 15sec, 30sec, 5sec et 9sec. D_5 envoi 4 paquets de 50 Bytes chacun et est donneur de 32sec.

En temps normal (sans attaques), après l'envoi du message de demande de jointure (Join Request), les noeuds sont enregistrés auprès du serveur. Le serveur gère le temps supplémentaire donné par le noeud donneur et le temps additionnel sera alloué en fonction de la stratégie de priorité (Voir Section 2.3.3.2) (par exemple, le serveur considère le noeud dédié à une application d'urgence comme étant le plus prioritaire). Dans notre cas, D_1 est le demandeur le plus prioritaire ("1"). Les autres noeuds demandeurs (D_2, D_3, et D_4) possèdent la même priorité ("2"), dans ce cas le demandeur ayant le plus faible trafic sera servi en premier (Voir Chapitre 4, Section 2.3.3.1).

Dans ce qui suit, l'objectif de l'évaluation est de montrer l'impact des attaques « Bit-flipping » et « ID-Spoofing » sur la QoS du réseau LoRaWAN en termes de PLR. En effet, dans notre cas, un attaquant peut produire une attaque sur un noeud afin d'arrêter son enregistrement sur le serveur avant de voler son identité à travers une attaque ID-Spoofing. Ce dernier falsifie la transmission d'un noeud en modifiant dans son paquet un octet sans avoir besoin de le déchiffrer en produisant une attaque Bit-flipping. Une fois le paquet attaqué arrive au serveur, et que le MIC soit vérifié, le paquet sera rejeté. Dans ce qui suit, nous présentons ce scénario où les deux attaques se produisent au même temps.

D'autant plus, l'AES est utilisé simplement pour la génération des clés. Le cryptage du paquet est effectué en utilisant l'opération XoR et non pas le cryptage AES. Ceci n'est pas suffisant pour garantir la sécurité, c'est ce qui permet aux attaques Bit-flipping et ID-Spoofing de se produire, modifiant ainsi certaines informations nécessaires au bon fonctionnement de E-LoRaWAN.

5.1 Scénario 1 : Bit-flipping et Requested Time-based ID-Spoofing

Dans notre scénario, durant le premier cycle les demandeurs (D_1, D_3 et D_4) sont satisfaits grâce au mécanisme de partage de duty cycle contrairement au demandeur D_2.

Étant donné que la probabilité qu'un noeud enregistré dans le réseau devient malveillant est très élevée [130], nous supposons que D_2 n'ayant pas profité des nouvelles fonctionnalités (Duty Cycle dynamique), devient malveillant durant le second cycle et commence par attaquer les noeuds afin de bénéficier de leurs services. Le nœud D_2 intercepte le $AppEUI$ de D_1 au

début de sa transmission et s'aperçoit qu'il s'agit d'une application importante. D_2 attaque D_1 en modifiant un octet de son paquet Join Request (remplace le 15 par 05 dans le temps demandé) pendant sa transmission (avant qu'il atteint le serveur). Ainsi, la demande de jointure envoyée par D_1 sera rejetée par le serveur lorsque ce dernier vérifiera le MIC. Tel qu'il est défini dans la Figure V.3, D_2 effectue l'attaque « Bit-flipping » pour empêcher D_1 d'envoyer son paquet de sorte qu'il ne puisse pas bénéficier du temps supplémentaire, le serveur rejette D_1, et n'aura pas à le conserver dans sa mémoire (D_1 n'est pas enregistré auprès du serveur). En outre, D_2, en produisant l'attaque « ID-Spoofing », il volera l'*AppEUI* de D_1 (*App*1) pour qu'il soit sur de pouvoir bénéficier du temps supplémentaire avant les autres noeuds. Le serveur ne pourra pas savoir que D_2 a non seulement bloqué D_1, ni qu'il a usurpé son identité.

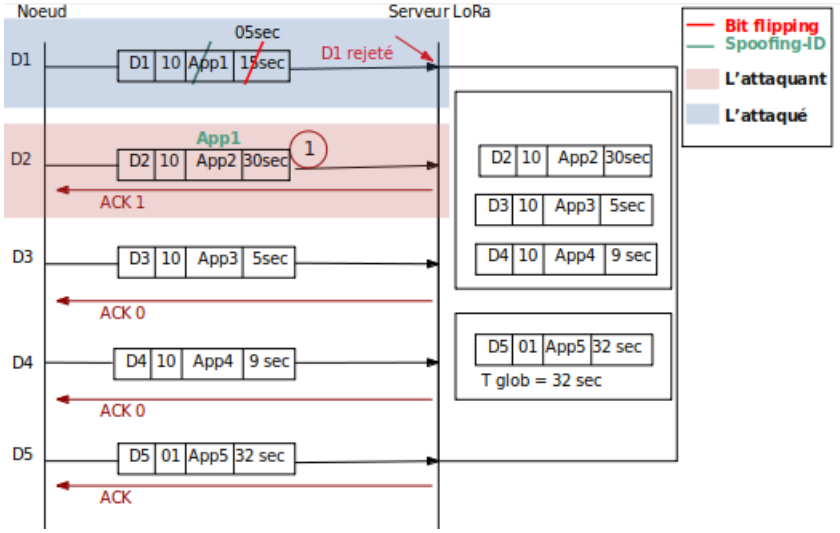


FIGURE V.3 – Scénario 1 : impact des attaques Bit-flipping et Priorité(ID)-Spoofing

Nous remarquons que la suspension de la transmission de D_1 suite à l'attaque Bit-flipping (en rouge dans la Figure V.3), et l'usurpation de son $APPEUI$, indiquant l'appartenance à une application d'urgence (App1) (en vert dans la Figure V.3) causée par le noeud malveillant D_2 a privé le noeud D_1 de s'enregistrer auprès du serveur et de bénéficier d'un temps partage de duty cycle. Ce dernier était le plus prioritaire, destiné à la transmission des messages d'alarme.

L'attaquant D_2 a pu, à travers les attaques produites, de bénéficier de l'envoi de ses paquets en priorité et de bénéficier d'un temps additionnel de $30sec$. De plus, du fait que la quantité de temps additionnel de D_2 soit très importante, les autres noeuds demandeurs n'ont pas pu avoir un ACK positif (ACK1) à leurs demandes de partage de Duty Cycle. Le nombre de noeuds satisfait a significativement diminué, seulement 40% des noeuds ont pu transmettre leurs paquets (PLR=60%).

5.2 Scénario 2 : Bit-flipping et RSSI based ID-Spoofing

Dans un second scénario, nous considérons que le noeud attaquant (D_2) a pour objectif de bénéficier du plus grand SF. Dans ce cas-là, D_2 va établir une attaque de bit-flipping en

changeant un octet de D_1 pour empêcher la réception de son paquet et ensuite continuer avec une attaque de ID-Spoofing afin d'usurper cette fois-ci l'identité de D_1 ayant le plus faible RSSI (demandant le plus grand SF). L'attaque ID-Spoofing est produite entre la gateway et le serveur, en interceptant l'information du RSSI rajoutée par la gateway.

Dans ce scénario, D_1 correspond au noeud le plus distant (faible RSSI). D_2 modifie un octet de D_1 (octet de AppEUI) afin que le MIC calculé au niveau du serveur ne sera pas le même que celui calculé au niveau du noeud et que le paquet soit rejeté par le serveur. Ensuite, D_2 ayant déjà eu l'information indiquant que D_1 est le noeud le plus distant à travers l'interception de son RSSI mesuré par les gateways, va usurper la mesure RSSI de D_1.

Dans ce scénario d'attaque, la QoS du réseau va être dégradée en termes de nombre de noeuds satisfaits et de consommation énergétique. D'une part, en effectuant l'attaque bit-flipping, D_1 ne pourras pas envoyer ses paquets et ceci résultera en une augmentation du PLR (diminuant le nombre de client satisfaits). D'autre part, l'attaque RSSI based ID-Spoofing a permis à $D2$ de profiter du $mode1$ au lieu du $mode9$, et donc de sur-utiliser des paramètres de configuration causant une sur-consommation énergétique.

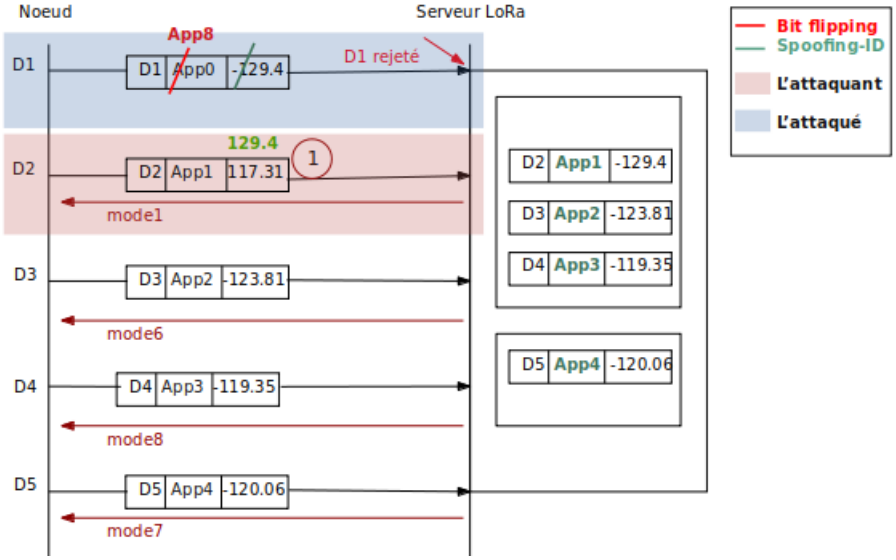


FIGURE V.4 – Scénario 2 : impact des attaques Bit-flipping et RSSI(ID)-Spoofing

6 – Solution contre la vulnérabilité de E-LoRaWAN

Contrairement aux travaux cités dans la littérature, proposant des solutions pour la génération des clés AES et pour l'authentification, nous proposons un nouveau schéma de cryptage comme étant une contre-mesure pour les attaques « Bit-flipping et ID-Spoofing ». L'idée principale de notre proposition est de rendre confidentiels les IDs des noeuds et des applications auxquelles chaque noeud est dédié, cela se fait en cryptant le message « Join Request » en utilisant la clé $AppKey$. Le cryptage du message permettra dans notre cas à E-LoRaWAN de lutter contre l'attaque « ID-Spoofing ». Ainsi le noeud malveillant ne peut plus récupérer les informations d'identité des noeuds normaux.

Nous proposons d'utiliser un mode de cryptage OCB-AES non-vulnérable à l'attaque « Bit-flipping » en raison de l'utilisation de la fonction HASH en plus du mélange de l'ordre des octets dans les messages lors de son cryptage. Ceci ne permet plus aux noeuds malveillants d'avoir connaissance du format du message crypté. Plusieurs études ont confirmé que le mode CTR utilisé par LoRaWAN est vulnérable aux attaques de type « bit flipping » [111, 126], contrairement au mode OCB connu pour être non vulnérable à cette attaque [127]. Nous présenterons dans les deux sous-sections, les deux solutions complémentaires l'une à l'autre afin de garantir la sécurisation de notre approche contre la combinaison des attaques « Bit-flipping et ID-Spoofing ».

6.1 Le message « Join Request » crypté

La Figure V.5 décrit notre schéma proposé pour sécuriser le message « Join Request » qui permet les mêmes exigences de sécurité de LoRaWAN (Authentification, intégrité, confidentialité) (Voir Section 3.4.5). En outre, notre schéma permet de prévenir l'usurpation d'identité dans le cadre du mécanisme de partage de Duty Cycle. Dans LoRaWAN une clé AppKey est enregistrée au niveau du noeud et du serveur avant la jointure "secret commun" et celle-ci est différente pour chaque noeud dans le cas d'une procédure OTAA, mais n'est pas utilisé par le noeuds lors de la jointure. Dans le schéma que nous proposons, le message de demande de jointure sera envoyé au serveur du réseau crypté en utilisant cette clé AppKey. Ainsi, en utilisant cette clé et avec le cryptage du Join Request, les IDs des noeuds (Application ID et Device ID) sont secrets et sécurisés. Aucun noeud ne pourra décrypter le Join Request d'un autre noeud n'ayant pas l'AppKey spécifique à chaque noeud et aucun noeud n'aura un aperçu des informations contenues dans le Join Request.

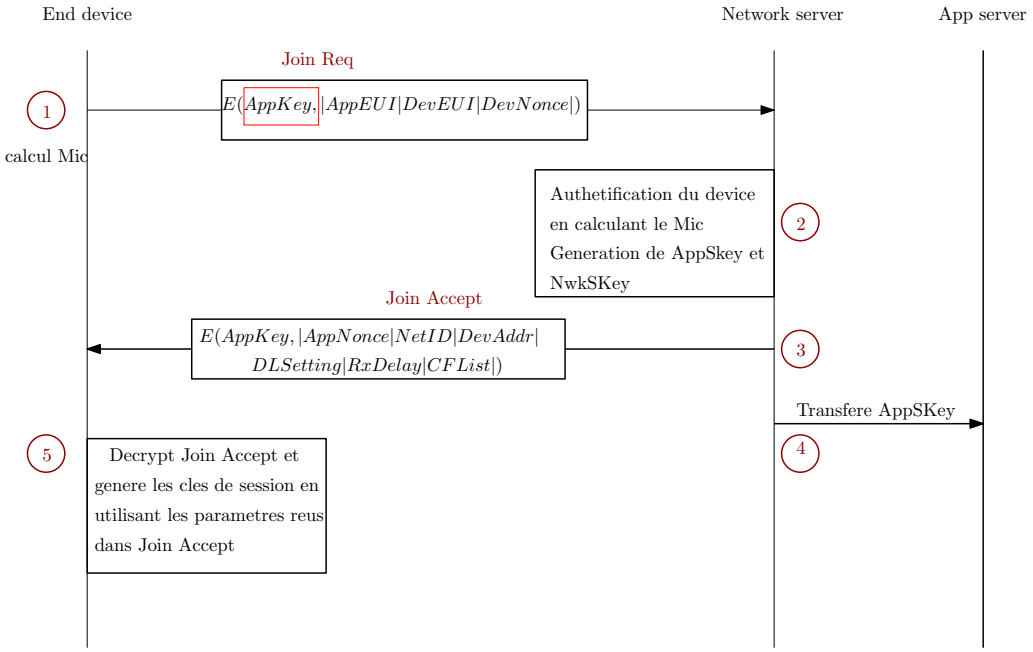


FIGURE V.5 – Join Request sécurisé

6.2 Offset Codebook Mode OCB-AES pour LoRaWAN

Certains modes AES sont accompagnés d'une preuve de sécurité basée sur l'hypothèse que le bloc de cryptage sous-jacent se comporte comme un générateur pseudo-aléatoire. La confidentialité et l'authenticité du message lors de l'utilisation d'un mode AES donné sont prouvées simultanément, et ce mode n'augmente pas de manière significative les possibilités de distinguer un texte chiffré d'un flux aléatoire en utilisant l'attaque de « Bit-flipping ». Le mode AES le plus sûr et le plus efficace contre cette attaque est le mode OCB. Le mode OCB est remarquable parce qu'il permet d'obtenir un cryptage authentifié en presque autant de temps que le mode conventionnel le plus rapide, contrairement au mode CTR qui ne permet que la confidentialité [128]. L'utilisation du mode OCB assure l'authentification et la confidentialité, ce mode est simple, propre et lutte contre plusieurs attaques telles que « Bit-flipping » en raison de l'utilisation d'une fonction HASH générant une clé HASH qui est incluse dans le cryptage et le décryptage des données secrètes connues uniquement par l'émetteur et le récepteur. L'algorithme 5 décrit le chiffrement OCB [129], et la Figure V.6 fournit une schéma d'accompagnement.

Algorithme 5 : Cryptage OCB [129]

input : K (Key), M (Message)
output : C (Cipher text)

1 Partition M into $M[1]$ ..... $M[m]$
2 $L \leftarrow AES(0^n)$
3 **for** $i \leftarrow 1$ **to** m **do**
4 $\quad Z[i] = L[i] \oplus DevNonce$
5 **end**
6 **for** $i \leftarrow 1$ **to** $m-1$ **do**
7 $\quad C[i] \leftarrow AES(M[i] \oplus Z[i]) \oplus Z[i]$
8 **end**
9 $X[m] \leftarrow (len(M[m]) \oplus L[-1]) \oplus Z[m]$
10 $Y[m] \leftarrow AES(X[m])$
11 $C[m] \leftarrow Y[m] \oplus M[m]$
12 $Hash \leftarrow M[1] \oplus ..... \oplus M[m-1] \oplus C[m] \oplus Y[m]$
13 $Tag \leftarrow AES(Hash \oplus Z[m])$
14 **Return** $C[1].....C[m]Tag$

La première étape du cryptage OCB consiste à partitionner le message M en m blocs de 128 bits (Ligne 1). Si le dernier bloc de message ne contient pas 128 bits, il est alors complété par des zéros. Nous pouvons ajouter au message un nombre de zéros allant de 1 à 127. Selon l'algorithme 5, le message crypté résulte en une concaténation des m blocs cryptés et du message Tag (Ligne 14), qui est la donnée secrète calculée à partir de la fonction HASH (Ligne 12). L'un des paramètres importants est la valeur L, qui diffère pour chaque bloc de messages. Cette valeur augmentera la complexité de la méthode de cryptage. Nous prenons 128 bits de 0 pour générer la valeur L et le nombre de valeurs L est égal au nombre de blocs de messages, ces valeurs sont générées comme il est décrit dans l'algorithme 6, où $ntz(i)$ est le nombre de

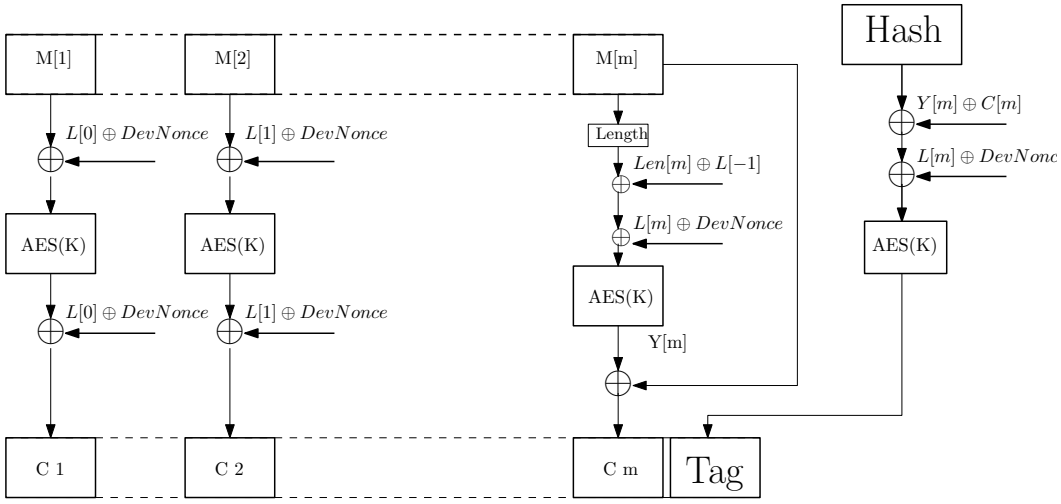


FIGURE V.6 – Mode OCB

zéros lorsque le nombre i est représenté en binaire.

Algorithme 6 : Génération des valeurs L [129]

1 $L[0]$ is the encrypted value, $L[0] \leftarrow AES(0^{128})$
2 $L[-1] = lsb(L[0]) \oplus Const43$
3 **for** $i \leftarrow 1$ **to** m **do**
4 $\quad L[i] \leftarrow L[i-1] \oplus len(ntz(i))$
5 **end**

Ensuite, DevNonce est utilisé pour générer la valeur Offset $Z[i]$. Pour chaque Offset $Z[i]$ la valeur L est Xorée au DevNonce (Ligne 4 dans l'algorithme 5). Cette valeur Offset est Xorée au message avant et après le cryptage AES (Ligne 7 dans l'algorithme 5). Dans le bloc de cryptage AES, chaque bloc de message effectue les opérations suivantes :

— Transformer le bloc en une matrice d'octets.

— Décaler les lignes.

— Mélanger les colonnes.

Dans chaque bloc de message, nous transformons les 128 bits en octets et nous les mettons sous la forme d'une matrice 16×16. Nous passons ensuite au décalage des rangées, de sorte que les éléments de la première rangée d'octets sont non décalés, les seconds sont décalés d'un octet vers la gauche, les éléments de la troisième ligne d'octets sont décalés de deux octets vers la gauche et les éléments de la dernière rangée d'octets sont décalés de trois vers la gauche. Enfin, nous avons une transformation qui opère sur chaque colonne, chaque élément de la matrice sera la somme des produits des éléments d'une ligne et d'une colonne. Chaque bloc de message résulte en un bloc de texte chiffré après le bloc AES(k). Grâce à cette procédure AES (décalage et mélange des octets), les noeuds malveillants n'auront aucune information sur le format des messages chiffrés.

La fonction HASH consiste à Xorer toutes les valeurs du message les unes aux autres, à l'exception du bloc final (Ligne 12 de l'algorithme 5). Cette fonction est utilisée pour l'authentification et comprend une donnée secrète qui permet de lutter contre une attaque de type « Bit-flipping ». L'étape finale consiste à générer le message chiffré qui est la concaténation de tous les blocs chiffrés ainsi que le Tag.

Dans la prochaine section, nous évaluons E-LoRaWAN en utilisant le schéma de sécurité proposé dans cette section.

7 – Évaluation du schéma de sécurité proposé pour LoRaWAN

En effet, en renforçant l'algorithme de cryptage LoRaWAN, notre schéma de sécurité contribue dans l'amélioration de la QoS du réseau en termes de PLR puisqu'il empêche la modification de l'information et la renforce contre les attaques « Bit-flipping et ID-Spoofing », permettant de profiter des éventuelles nouvelles fonctionnalités du protocole E-LoRaWAN proposé précédemment. Cependant, ce renforcement nécessiterait une augmentation du temps de traitement et donc la consommation énergétique. Dans cette partie, nous évaluons notre schéma proposé en termes de temps de traitement et de consommation énergétique et nous discutons cette augmentation afin de démontrer la faisabilité de notre proposition.

Pour évaluer les performances du schéma de sécurité proposé, nous avons réalisé une expérimentation en utilisant les modules LoRa SX1272 waspmote de Libelium et STM32 de STMicroelectronics.

7.1 Évaluation du délai de transmission

Notre nouveau schéma de cryptage consiste en un traitement supplémentaire au niveau du noeud et du serveur comparé au schéma original de LoRaWAN [21], ceci provoque une incrémentation du temps de cryptage. Ainsi, nous proposons de comparer le délai de la procédure de jointure utilisant le nouveau schéma et le schéma original LoRaWAN.

Le délai de la procédure de jointure est mesuré depuis le début de transmission du Join Request jusqu'à la fin de la procédure de jointure lorsque le noeud reçoit le Join Accept. Les mesures sont présentées dans la Table V.1. Selon ce résultat, le délai de la procédure de jointure utilisant le nouveau schéma de sécurité est augmenté d'environs $35ms$.

Table V.1. Délai total de la procédure de jointure

	Schéma LoRaWAN	Nouveau schéma
Délai de la procédure de jointure	5912.57 ms	5947.28 ms

En effet, notre schéma consiste en deux nouvelles contributions pour la procédure de jointure causant cette augmentation : l'ajout du cryptage du Join Request, et le changement du mode de cryptage. Afin de mieux comprendre la cause impactant le plus sur l'augmentation du délai de la procédure de jointure, nous avons évalué le délai de traitement après chaque étape de la procédure de jointure : ToA du Join Request (ToA_{JR}), Temps de traitement du Join Request par le serveur ($Tproc_{JR}$), ToA du Join Accept (ToA_{JA}), Temps de traitement

du Join Accept par le noeud ($Tproc_{JA}$). La Table V.2 présente le temps de traitement des différentes étapes de la procédure de jointure.

Table V.2. Délais des différentes étapes de la procédure de jointure

	ToA_{JR}	$Tproc_{JR}$	ToA_{JA}	$Tproc_{JA}$
Schéma LoRaWAN	1481.91 ms	1332.27 ms	1155.07 ms	1943.32
Nouveau Schéma	1488.67 ms	1350.1 ms	1159.41	1948.47 ms

D'après les résultats obtenus dans la Table V.2, ToA_{JR} et $Tproc_{JA}$ correspondants à respectivement : ToA du Join Request et ToA du Join Accept ont augmenté par rapport à ceux correspondant au schéma original de LoRaWAN. Cette augmentation se traduit par l'ajout de la fonction HASH (sur 2 Bytes) dans l'algorithme de cryptage OCB. En outre, d'une part, la durée de traitement $Tproc_{JA}$ a augmenté en utilisant le nouveau schéma du fait que le mode de cryptage AES-OCB consiste en un remaniement des octets de la charge utile avant son cryptage, nécessitant plus de délai. D'autre part, $Tproc_{JR}$ a augmenté d'une marge plus importante que celle observée pour $Tproc_{JA}$. Cette grande marge revient au fait que dans le schéma LoRaWAN original, le Join Request est en clair et est traité directement au niveau serveur sans avoir recours au décryptage.

En résumé, le délai de la procédure de jointure de notre nouveau schéma a augmenté d'environs $35ms$. Cependant, le réseau n'est plus confronté aux modifications d'information et bénéficie d'une atténuation des pertes. Le nouveau schéma proposé reste faisable en termes de délai comparé au délai de la procédure de jointure lorsqu'une modification de l'information se produit dans le cas d'un schéma original de LoRaWAN, provoquant un rejet de la procédure de jointure, induisant d'autres tentatives de jointure (retransmissions du Join Request), et donc une augmentation du délai présenté dans la Table V.1.

7.2 Évaluation de la consommation énergétique

La consommation énergétique des noeuds est évalué en utilisant le nouveau schéma proposé et sera comparée à la consommation énergétique du schéma original. La Table V.3 présente les résultats obtenus. Cette consommation énergétique est mesurée en utilisant l'Eq III.2 [87].

Table V.3. Consommation énergétique de la procédure de jointure (J)

	Schéma LoRaWAN	Nouveau schéma
Cryptage du JR	Non crypté	0.017
Décryptage du JA	0.195	0.209

Les résultats obtenus montrent une légère augmentation de la consommation énergétique dans le cas d'utilisation du nouveau schéma de sécurité. En effet, l'ajout du cryptage du Join Request augmente la consommation d'énergie au niveau du noeud de $0.017J$. D'autre part, le processus de décryptage du Join Accept nécessite aussi plus d'énergie. Cependant, comme mentionné dans la section précédente, l'utilisation du Schéma LoRaWAN original ne garantit pas une réception et acceptation de la jointure, ce qui nécessite des retransmissions du Join Request, et donc plus de consommation énergétique.

8 – Conclusion

Dans ce chapitre, nous avons ouvert une discussion sur la vulnérabilité du protocole proposé E-LoRaWAN face à deux attaques : « Bit-flipping et ID-Spoofing ». Nous avons également discuté les mesures de sécurité pouvant être prises. Nous avons proposé deux solutions combinées : la première pour protéger les informations du Join Request en utilisant la clé AES128 $AppKey$ pour son cryptage et la seconde était d'utiliser le mode de cryptage OCB-AES qui consiste à mélanger et décaler les positions des octets du paquet LoRa en plus des opérations Xor avec les clés générées et de sa concaténation à une clé secrète HASH. Le schéma que nous proposons permet de mieux sécuriser le protocole et de maximiser la satisfaction des noeuds demandeurs en commençant par les noeuds ayant la plus haute priorité en fonction de la stratégie adoptée par le serveur LoRa. Les résultats obtenus permettent de confirmer l'efficacité et la faisabilité de notre solution de sécurité en termes de délai et de consommation énergétique. Ce chapitre montre qu'en effet, d'autres défis restent à relever en regards du pilier sécurité de notre proposition "E-LoRaWAN" et présente une ouverture vers des perspectives pour nos futurs travaux de recherche.

Conclusion générale et perspectives

L'accroissement du déploiement des environnements IoT et l'utilisation des applications IoT dans différents domaines fait face au principal défi "amélioration de la qualité de service (QoS)" en termes de fiabilité et d'efficacité énergétique. Pour répondre à ce défi, l'IoT s'appuie sur l'utilisation d'un ensemble de technologies LPWAN (Low Power Wide Area Networks), notamment grâce à ses caractéristiques de longue portée, faible débit et faible consommation d'énergie. Les travaux développés dans le cadre de cette thèse ont justement pour but de relever certaines problématiques liées à la gestion de QoS, en exploitant les performances d'une des technologies LPWAN, LoRa et son protocole MAC LoRaWAN. L'objectif est de proposer des solutions permettant d'un côté de répondre au besoin en termes de QoS et d'un autre côté d'optimiser les ressources ainsi que la consommation énergétique.

Nous avons d'abord entamé nos travaux par une étude critique sur les principaux types de réseaux existants. Nous avons abordé une discussion sur les caractéristiques et le fonctionnement de base des normes LPWAN ainsi que les avantages et les inconvénients de certaines normes de ce type de réseau. Nous nous sommes intéressés par la suite à la norme LoRa, notamment son protocole MAC (LoRaWAN) ouvert. Nous avons identifié des verrous à lever pour le protocole LoRaWAN concernant l'allocation des ressources et l'accès au canal.

Dans un premier travail, nous nous sommes intéressés principalement à l'amélioration du mécanisme d'adaptation de débit (ADR) proposé par LoRaWan afin de gérer individuellement la configuration de chaque noeud pour lui offrir le meilleur débit à utiliser.

Après avoir fait une étude de l'art des différentes propositions publiées dans la littérature liées à l'adaptation de débit, nous avons remarqué que ces solutions n'ont pas pris en charge le cas de noeuds IoT mobiles. Dans ce cadre, notre première contribution a porté sur la définition d'une amélioration du mécanisme d'adaptation de débit (E-ADR) dans un contexte de mobilité connu. E-ADR offre une solution de calcul d'une nouvelle configuration à offrir aux noeuds mobiles en se basant sur leurs futurs déplacements définis à travers une méthode de régression linéaire et un encadrement des valeurs RSSI correspondantes par rapport au meilleur débit à offrir. Les résultats de simulation effectués pour différents scénarios ont montré que E-ADR est meilleur que les autres variantes ADR en termes de PLR. En effet, E-ADR réduit considérablement (voir élimine dans certains cas) les pertes dues à un mauvais choix de SF tout en améliorant l'efficacité énergétique.

Ensuite, une amélioration a été portée à ce mécanisme pour tenir compte d'un contexte de mobilité indéfini. Pour ce faire, une technique de prédiction de la trajectoire des nœuds basée sur le modèle de Markov caché a été exploitée pour définir la position du nœud mobile et décider de la meilleure configuration assurant une meilleure QoS. Ce mécanisme baptisé VHMM-based E-ADR offre un bon compromis entre la fiabilité et l'efficacité énergétique. Les études de simulation effectuées ont validé dans une première partie la prédiction à travers le test d'un modèle de trajectoire connu, ensuite elle s'est intéressé à l'évaluation de VHMM-based

E-ADR en le comparant aux autres variantes. Les résultats ont confirmé le dépassement de VHMM-based E-ADR aux autres variantes en termes de PLR et Consommation énergétique.

Dans une deuxième partie, nous avons constaté que souvent les pertes sont dues au dépassement de la restriction du Duty Cycle. Ainsi, notre deuxième contribution dans cette thèse est la proposition d'un mécanisme de distribution dynamique du Duty Cycle, portant sur l'emprunt du Duty Cycle non consommé et non-utilisable par un noeud donné afin d'autoriser à un autre de dépasser son Duty Cycle. Ce mécanisme a permis d'augmenter le nombre de paquets envoyés par chaque noeud du réseau améliorant ainsi la QoS globale. Son implémentation avec ADR basique ou même E-ADR améliore les performances en réduisant davantage les pertes dues au dépassement de duty cycle.

Bien que la proposition de la solution E-ADR permet une meilleure fiabilité grâce à la stratégie de distribution optimale de ces paramètres sur l'ensemble des noeuds du réseau LoRa, un déploiement dense peut engendrer des collisions des transmissions simultanées utilisant les même paramètres de configuration sur le même canal choisi aléatoirement. Notre troisième contribution dans cette thèse a porté sur la proposition d'un mécanisme d'accès déterministe basé sur l'ordonnancement des slots d'une manière dynamique selon le principe TDMA avec une allocation de canaux FIFO. Cette contribution a été baptisée "Approche triplement conjointe SF-Slot-Canal" permet d'éviter les collisions, évitant ainsi les retransmissions des paquets et donc offrant une meilleure efficacité énergétique. L'ensemble de nos contributions peut être implémenté au niveau de LoRaWAN donnant lieu à un nouveau protocole baptisée E-LoRaWAN.

Dans une dernière contribution, nous avons étudié la vulnérabilité de notre proposition E-LoRaWAN contre certaines attaques. Nous avons examiné le risque des attaques Bit-flipping et ID-Spoofing et les mesures pouvant être prises dans le protocole E-LoRaWAN contre ces attaques. Nous avons conçu un schéma de sécurité se basant sur l'utilisation d'une clé partagée pour le cryptage du Join Request évitant ainsi l'usurpation d'identité des noeuds par l'attaque ID-Spoofing. Nous avons aussi proposé l'utilisation du mode de cryptage AES-OCB à la place d'une simple opération XoR effectuée par le mode CTR dans le protocole de base LoRaWAN. Le nouveau schéma consiste en une méthode de protection du cryptage des paquets basée sur le remaniement de l'emplacement des octets, permettant la transmission sûre des données et une protection contre l'attaque ID-Spoofing. Ce schéma permet de protéger les noeuds bénéficiant des nouvelles fonctionnalités du E-LoRaWAN.

En résumé, nous avons pu, moyennant les différentes solutions présentées, proposer une amélioration du protocole LoRaWAN baptisée « E-LoRaWAN », dédié au support de mobilité. Ce dernier permet d'assurer une bonne fiabilité tout en garantissant une bonne efficacité énergétique, ceci en assurant une configuration adéquate, une gestion et une optimisation efficace des ressources, et moins de vulnérabilité.

Les contributions présentées dans cette thèse ouvrent plusieurs perspectives pour des travaux futurs, à savoir dans l'axe de sécurité ou d'optimisation de performance. Nous mettons en évidence quelques perspectives principales.

Les travaux proposés dans notre thèse concernent la performance pour les communications montantes (UL) dans le réseau LoRaWAN. Cependant, cette performance est aussi limitée par les communications descendantes allant des gateways aux noeuds LoRa [131–133]. En effet, le lien descendant est limité par plusieurs facteurs et plusieurs problématiques restent

non résolues : en effet, La gateway est limitée par un seul envoi à un instant donné et cet envoi ne peut se faire que pendant deux fenêtres de réception (Classe A) pour une meilleure efficacité énergétique. Cependant, ces transmissions peuvent être sujettes à des collisions avec des transmissions montantes. Une coordination plus efficace entre la voix montante et la voix descendante est nécessaire. Par ailleurs, dans le cas de présence de plusieurs gateways, le serveur réseau LoRa se limite au choix de la gateway qui offre la meilleure qualité de signal pour un noeud donné. Une meilleure coordination entre les gateways avec une technique d'équilibrage de charge pourrait améliorer la performance globale du réseau. En effet, tout comme les noeuds, les gateways sont limitées par un duty cycle pouvant limiter leurs transmissions DL, particulièrement lorsque le nombre de noeuds est important.

En outre, comme mentionnés dans notre cinquième chapitre, les problèmes liés à la sécurité freinent considérablement l'évolution et le déploiement rapide de la technologie LoRa et son protocole LoRaWAN. Les attaques par déni de service (DoS) représentent un vrai danger pour le réseau LoRaWAN, notamment le protocole proposé E-LoRaWAN. Une étude plus approfondie des problèmes de sécurité est nécessaire pour trouver des solutions limitant les problèmes d'attaques, mais également qui s'adaptent aux ressources limitées des objets IoT.

Liste des publications

— Norhane Benkahla, Hajer Tounsi, Ye-Qiong Song, Mounir Frikha : «Enhanced Dynamic Duty Cycle in LoRaWAN Network». ADHOC-NOW 2018 : 147-162.

— Norhane Benkahla, Boutheyna Belgacem, Mounir Frikha : «Security analysis in Enhanced LoRaWAN Duty Cycle». COMNET 2018 : 1-7.

— Norhane Benkahla, Hajer Tounsi, Ye-Qiong Song, Mounir Frikha : «Enhanced ADR for LoRaWAN networks with mobility». IWCMC 2019 : 514-519.

— Norhane Benkahla, Hajer Tounsi, Ye-Qiong Song, Mounir Frikha : «Review and evaluation of ADR enhancements for LoRaWAN networks». Telecommunication systems Journal (TELS 2020, IF 1.734). (Date d'acceptation : 27/10/2020).

Bibliographie

[1] D. Patel, and M. Won. "Experimental study on low power wide area networks for mobile internet of things". In : the 85th Vehicular Technology Conference (VTC Spring). Sydney, Australia. pp 1–5. (2017).

[2] T. Attia, M. Heusse, B. Tourancheau, and A. Duda. "Experimental Characterization of Packet ReceptionRate in LoRaWAN,". In : Rencontres Francophones sur la Conception de Protocoles, l'Évaluation de Performance et l'Expérimentation des Réseaux de Communication. France. (2019)

[3] The Thing Network Wiki. "Adaptive Data Rate". Available : https ://www.thethingsnetwork.org/wiki/LoRaWAN/ADR. (2017).

[4] M. Slabicki, G. Premsankar, and M. Di Francesco. "Adaptive configuration of loRa networks for dense IoT deployments". In : Network Operations and Management Symposium. Taiwan. (2018).

[5] F. Cuomo, Ma. Campo, A. Caponi, G. Bianchi, G. Rossini, and P. Pisani. "EXPLoRa : Extending the performance of LoRa by suitable spreading factor allocations". In : the 13th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob). Italy. (2017).

[6] B.S. Chaudhari, and M. Zennaro. "LPWAN Technologies for IoT and M2M Applications". In : Elsevier. London, UK. ISBN 978-0-12-818880-4. (2020).

[7] U. Raza, P. Kulkarni, and M. Sooriyabandara. "Low Power Wide Area Networks : An Overview". In : IEEE Communications Surveys Tutorials. Vol 19. No 2. pp 855-873. Doi : 10.1109/COMST.2017.2652320. (2017).

[8] M. Centenaro, L. Vangelista, A. Zanella, and M. Zorzi. "Long-range communications in unlicensed bands : The rising stars in the IoT and smart city scenarios". In : IEEE Wireless Communications Journal. Vol 23. pp 60–67. Doi : 10.1109/MWC.2016.7721743. (2016).

[9] W. Tong, P. Zhu. "Huawei whitepaper - 5G : A Technology Vision". (2014).

[10] A.M. Baharudin, and W. Yan. "Long-range wireless sensor networks for geolocation tracking : Design and evaluation". In : the International Electronics Symposium (IES). Indonesia. pp 76–80. (2016).

[11] W. Guibene, J. Nowack, N. Chalikias, and M. Kelly. "Evaluation of LPWAN technologies for smart cities : River monitoring use-case". In : IEEE Wireless Communications and Networking Conference Workshops (WCNCW). San Francisco CA USA. pp. 17–22. (2017).

[12] O. Vondrous, Z. Kocur, T. Hegr, and O. Slavicek. "Performance evaluation of IoT mesh networking technology in ISM frequency band". In : the 17th International Conference on Mechatronics - Mechatronika (ME). Prague Czech Republic. pp 1–8. (2016).

[13] M. Weyn, G. Ergeerts, R. Berkvens, B. Wojciechowski, and Y. Tabokov. "Dash7 alliance protocol 1.0 : Low-power, mid-range sensor and actuator communication". In : IEEE Conference on Standards for Communications and Networking (CSCN). Tokyo Japan. (2015).

[14] Weightless sig. Available :http ://www.weightless.org/.

[15] Sigfox. Available : https ://www.sigfox.com/en.

[16] E. D.Poorter, J. Hoebeke, M. Strobbe, I. Moerman, S. Latré, M. Weyn, B. Lannoo, and J. Famaey. "Sub-GHz LPWAN network coexistence, management and virtualization : an overview and open research challenges". In : Wireless Personal Communications Journal. pp 187-213. (2017).

[17] 3gpp low power wide area technologies - gsma white paper. Available : https ://www.gsma.com/iot//wp-content/uploads/2016/10/3GPP-Low-Power-Wide-Area-Technologies-GSMA-White-Paper.pdf.

[18] A. Adhikary, X. Lin, and Y. P. Wang. "Performance evaluation of nb-iot coverage". In : IEEE 84th Vehicular Technology Conference (VTC-Fall). Montreal Canada. (2016).

[19] Y. P. Wang, X. Lin, A. Adhikary, A. Grovlen, Y. Sui, Y. Blankenship, J. Bergman, and H. S. Razaghi. "A primer on 3gpp narrowband internet of things (nb-iot)". In : IEEE Communications Magazine. Vol. 55. pp 117-123.

[20] LoRa Alliance. Lora-Alliance. Available : https ://www.lora-alliance.org. (2017).

[21] N. SORNIN (Semtech), and A. YEGIN (Actility). LoRaWAN Specification v1.1. (2017).

[22] P. Neumann, J. Montavont, and T. Noel. "Indoor deployment of low-powerwide area networks (lpwan) : A lorawan case study". In IEEE 12th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob). New York USA. pp 1–8.

[23] K. Mikhaylov, T. Haenninen. "Analysis of capacity and scalability of the LoRa low power wide area network technology". In : European Wireless 2016 ; 22th European Wireless Conference. Oulu Finland. pp. 119–12. (2016).

[24] Sigfox world coverage. Available : www.sigfox.com/en/coverage/.

[25] LoRa world coverage. Available in : www.lora-alliance.org/.

[26] R. Sinha, Y. Wei, S. Hwang. "A survey on LPWA technology : LoRa and NB-IoT". ICT Convergence Technology Journal. Vol 3. pp 14–21. (2017).

[27] H.S. Dhillon, H. Huang, H. Viswanathan. "Wide-area wireless communication challenges for the Internet of Things". In : IEEE Communications Magazine. pp 168-174. (2016).

[28] NB-IOT Enabling New Business Opportunity. Available : https ://www.huawei.com/minisite/4-5g/img/NB-IOT.pdf.

[29] T.G. Durand. "Evaluation of Next Generation Low-Power Communication Technologies to Replace GSM in IoT Applications". Master's Thesis, Department of Electrical and Electronic Engineering, Stellenbosch University, Stellenbosch, South Africa. (2018).

[30] H. Wang, A.O. Fapojuwo. "A survey of enabling technologies of low power and long range machine to machine communications". In : IEEE Communications Surveys Tutorials. pp 2621–2639. (2017).

[31] G.A Akpakwu, B.J. Silva, G.P. Hancke, A.M. Abu-Mahfouz. "A survey on 5G networks for the internet of things : Communication technologies and challenges". In : IEEE Access. pp 3619–3647. (2017).

[32] NB-IoT Tutorial-Features, Spectrum, Applications of NB-IoT. Available : http ://www.rfwireless-world.com/Tutorials/NB-IoT-tutorial.html

[33] D. Magrin, M. Centenaro, and L. Vangelista. "Performance evaluation of LoRa networks in a smart city scenario". In : IEEE International Conference on Communications (ICC). Paris France. pp 1-7. (2017).

[34] M. Bor, and U. Roedig. (2017). "LoRa transmission parameter selection". In : the 13th International Conference on Distributed Computing in Sensor Systems (DCOSS). Ottawa Canada. pp 27-34. (2017).

[35] M. Bor, J. Vidler, and U. Roedig. "LoRa for the Internet of Things". In : the International Conference on Embedded Wireless Systems and Networks (EWSN). Austria. pp 361-366. (2016).

[36] C. Goursaud, and J-M. Gorce. "Dedicated networks for IoT : PHY/MAC state of the art and challenges". In : the endorsed transactions on Internet of Things (EAI). Vol 1, pp 1-11. (2015).

[37] D. Croce, M. Gucciardo, I. Tinnirello, D. Garlisi, and S. Mangione. "Impact of spreading factor imperfect orthogonality in loRa communications". In : the International Tyrrhenian Workshop on Digital Communication. Switzerland. pp 165-179. (2017).

[38] Libelium comunicaciones Distribuidas S.L. Waspmote LoRa 868 MHz 915 MHz SX1272 networking guide. (2015).

[39] Hope RF Microelectronics. RFM95/96/97/98(W)—Low Power Long Range Transceiver Module. China. (2016).

[40] LoRa Alliance Technical committee, LoRaWAN - Regional Parameters, 1.0, LoRa Alliance, Inc., Jul. 45 pp. (2016).

[41] ETSI. (May 2012). ETSI EN 300 220-1 V2.4.1 (2012-05). Available : www.etsi.org/deliver/etsi-en/300200-300299/30022001/02.04.01-60/ en-30022001v020401p.pdf.

[42] I. Butun, N. Pereira and M. Gidlund. "Analysis of LoRaWAN v1.1 security : research paper." In : the 4th ACM MobiHoc Workshop on Experiences with the Design and Implementation of Smart Objects. New York USA. pp 1-6. (2018).

[43] F. Ferrero. "LoRaWAN Mac Layer". (2019).

[44] A. Augustin, J. Yi, T. Clausen, W.M. Townsley. "A study of LoRa : Long range low power networks for the internet of things". Sensors. (2016).

[45] O. Alvear, J. H-Tapia, C. Calafate, E. H-Orallo, J-C Cano, and P. Manzoni. "Assessing the Impact of Mobility on LoRa Communications". In : InterIoT/SaSeIoT'17. Vol 242. Doi : 10.1007/978-3-319-93797-7-10. (2017).

[46] R. S-Cabrera, A. Pachon, and J-M. Madrid. "Proof of Concept of an IoT-Based Public Vehicle Tracking System, Using LoRa (Long Range) and Intelligent Transportation System (ITS) Services". In : Journal of Computer Networks and Communications. Vol 19. Doi : 10.1155/2019/9198157. (2019).

[47] K. Kousias, G. Caso, O. Alay, and F. Lemic. Empirical Analysis of LoRaWAN Adaptive Data Rate for Mobile Internet of Things Applications. In : Wireless of the Students, by the Students, and for the Students Workshop. Mexico. (2019)

[48] Semtech Corportation, Smart agriculture. Available : https ://www.semtech.com/lora/lora-applications/smart-agriculture.

[49] Libelium comunicaciones Distribuidas S.L. Waspmote LoRa 868 MHz 915 MHz SX1272 networking guide. (2015).

[50] D Y. Kim, S. Kim, H. Hassan, and J H. YukPark. "Adaptive data rate control in low power wide area networks for long range IoT services". In : Journal of computational science. Vol 22. pp 171-178. (2017).

[51] K.Q. Abdelfadeel, V. Cionca, and D. Pesch. "A fair adaptive data rate algorithm for loRaWAN". In : International Conference on Embedded Wireless Systems and Networks (EWSN). Spain. (2018).

[52] R.M. Sandoval, A.-J. Garcia-Sanchez, J. Garcia-Haro. "Performance optimization of LoRa nodes for the future smart city/industry". In : EURASIP Journal on Wireless Communications and Networking. Vol 200. Doi : 10.1186/s13638-019-1522-1. (2019).

[53] Bor, M.C. ; Roedig, U. ; Voigt, T. ; Alonso, J.M. Do LoRa Low-Power Wide-Area Networks Scale ? In Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, New York, NY, USA. pp. 59–67. (2016).

[54] J. Petaejaejaervi, K. Mikhaylov, M. Pettissalo, J. Janhunen, and J. Iinatti. "Performance of a low-power wide-area network based on lora technology : Doppler robustness, scalability, and coverage". In : the International Journal of Distributed Sensor Networks. Vol 13. No 3. pp 116. Doi : 10.1177/1550147717699412. (2017).

[55] D. Bankov, E. Khorov, and A. Lyakhov. "Mathematical model of lorawan channel access". In : 18th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM). Macau China. pp. 13. (2017).

[56] J. Petajajarvi, M. Pettissalo, K. Mikhaylov, A. Roivainen, T. Hänninen. "On the Coverage of LPWANs : Range Evaluation and Channel Attenuation Model for LoRa Technology". In : the International Conference on ITS Telecommunications (ITST), Copenhagen Denmark. (2015).

[57] O. Georgiou, U. Raza. "Low Power Wide Area Network Analysis : Can LoRa Scale ?". In : IEEE Wireless Communications Letters. pp 162–165. (2017).

[58] D. Magrin, M. Capuzzo and A. Zanella. "A Thorough Study of LoRaWAN Performance Under Different Parameter Settings". In : IEEE Internet of Things Journal. Vol 7. No 1. pp 116-127. (2020).

[59] D. Phan. "Contrôle de la puissance pour les réseaux sans fil". (2014).

[60] H. Kwasme and S. Ekin. "RSSI-Based Localization Using LoRaWAN Technology". In : IEEE Access. pp 99856-99866. doi : 10.1109/ACCESS.2019.2929212. (2019).

[61] P. Manzoni, C.T. Calafate, J.C. Cano, and E. Hernández-Orallo. "Indoor Vehicles Geolocalization Using LoRaWAN". In : Future Internet 2019. Doi : 10.3390/fi11060124. (2019).

[62] N.M. Drawil, H.M. Amar, and O.A. Basir. "Gps localization accuracy classification : A context-based approach". In : IEEE Transactions on Intelligent Transportation Systems. Vol 14. No 1. pp 262–273. (2013).

[63] A. Varshavsky, M.Y. Chen, E. de Lara, J. Froehlich, D. Haehnel, J. Hightower, A.y LaMarca, F. Potter, T. Sohn, K. Tang, et al. "Are gsm phones the solution for localization ?". In : the 7th IEEE Workshop on Mobile Computing Systems and Applications (WMCSA). Orcas Island USA. pp. 34–42. (2005).

[64] Lucia J Zamorano, Lutz Nolte, A Majeed Kadi, and Zhaowei Jiang. "Interactive intraoperative localization using an infrared-based system" In : Minim Invasive Neurosurg. Vol 15. No 5. pp 290–298. (1993).

[65] J.L Rose, and P.B Nagy.(2000). "Ultrasonic waves in solid media". In : The Journal of the Acoustical Society of America". Vol 107. No 4. pp 1807–1808.

[66] Sigfox Geolocation, Available : http ://www.sigfox.com/en/sigfox-geolocation.

[67] H. Tsuji, P. Cherntanomwong, and J.-I. Takada. (2007). "Experiential evaluation of outdoor radio source localization using spatial information of array". In : Asia-Pacific Microwave Conference. Tailand.

[68] B.C. Fargas, and M.N. Petersen. "GPS-free geolocation using LoRa inlow-power WANs". In : Global Internet of Things Summit (GIoTS). Switzerland. (2017).

[69] Q.H. Spencer, B.D. Jeffs, M.A. Jensen, and A.L. Swindlehurst. "Modeling the statistical time and angle of arrival characteristics of an indoor multipath channel". In : IEEE Journal on Selected Areas in Communications. Vol 18. No 3. pp. 347–360. Doi : 10.1109/49.840194. (2000).

[70] S. Hamdoun, A. Rachedi, and A. Benslimane. "Comparative analysis of rssi-based indoor localization when using multiple antennas inwireless sensor networks". In : Proceedings of the International Conference on Selected Topics in Mobile and Wireless Networking (MoWNeT). Montreal. pp.146–151. (2013).

[71] E. Goldoni, A. Savioli, M. Risi, and P. Gamba. "Experimental analysis of RSSI-based indoor localization with IEEE 802.15.4". In : European wireless proceeding. Italy. pp 71–77. (2010).

[72] C. Saunders, A. Gammerman, and V. Vovk. "Ridge Regression Learning Algorithm in Dual Variables". In : the 15th Proceedings of International Conference on Machine LearningJuly (ICML). pp 515-521. (1998).

[73] Y. Ishikawa, Y. Tsukamoto, and H. Kitagawa. "Extracting mobility statistics from indexed spatio-temporal datasets". In : Spatio-Temporal Database Management Workshop (STDBM). Toronto Canada. pp 9–16. (2004).

[74] H. Jeung, H. Shen, and X. Zhou. "Mining trajectory patterns using hid-den Markov models". In : International Conference on Data Warehousing and Knowledge Discovery (DaWaK). Vol 4654. pp 470–480. (2007).

[75] Q. Peng, Z. Ding, and L. Guo. "Prediction of trajectory based on Markov chains". In : the International Conference on Computer Science and Information Technology (ICCSIT). pp. 189–193. (2010).

[76] T. Feng, Y. Guo, K. Huang, and J. Ji. "A behavior trajectory restoration algorithm based on hidden Markov models". In : Computer Science - Computer Engineering. Vol 38. No 12. pp 1–5. (2012).

[77] S. Gambs, M. Killijian, D. P. Cortez, and N. Miguel. (2012). "Next place prediction using mobility Markov chains". In : the First Workshop on Measurement, Privacy, and Mobility. pp. 1–6.

[78] L. Song, D. Kotz, R. Jain, and X. He. "Evaluating next-cell predictors with extensive wifi mobility data". In : IEEE Transactions on Mobile Computing. Vol 5. No 12. pp 1633–1649. (2006).

[79] A. B. Cheikh, M. Ayari, R. Langar, G. Pujolle, and L. A. Saidane. "Optimized handoff with mobility prediction scheme using hmm for femto cell networks". In : IEEE International Conference on Communications (ICC). pp 3448–3453. (2015).

[80] K. L. Yap and Y. W. Chong. "Optimized access point selection with mobility prediction using hidden markov model for wireless network". In : the 19th International Conference on Ubiquitous and FutureNetworks (ICUFN). pp 38–42. (2017).

[81] M. Petit, H. Christiansen. "Un calcul de Viterbi pour un Modèle de Markov Caché Contraint". Publié dans Cinquièmes Journées Francophones de Programmation par Contraintes, Actes JFPC 2009. Orléans, France. (2009).

[82] H. Zhang and L. Dai. "Mobility Prediction : A Survey on State-of-the-Art Schemes and Future Applications". In : IEEE Access. Vol 7. pp 802-822. (2019).

[83] Libelium Comunicaciones Distribuidas S.L. Waspmote data frame programming guide. (2017).

[84] Semtech Corporation, STMicroelectronics. Discovery kit for LoRaWAN, Sigfox, and LPWAN protocols with STM32L0 B-L072Z-LRWAN1. (2019).

[85] https ://www.digitalwallonia.be/fr/publications/smart-farming.

[86] U. Peura. "LoRaWAN optimization for a batery powered sensor network". Bachelor's Thesis. Oulu University of Applied Sciences. (2018).

[87] SX1272/73 : 860 MHz to 1020 MHz Low Power Long Range Transceiver. Available : http ://www.semtech.com/images/datasheet/sx1272.pdf.

[88] P. CongDuc. "QoS for long-range wirless sensors under duty-cycle regulation with shared activity time usage". In : ACM Transactions on Sensor Networks. Vol 12. No 4. Doi : 10.1145/2979678. (2016).

[89] J. T. Raj and J. Sankar. "IoT based smart school bus monitoring and notification system". In : IEEE Region 10 Humanitarian Technology Conference (R10-HTC). Dhaka Bangladesh. pp 89-92. Doi : 10.1109/R10-HTC.2017.8288913. (2017).

[90] LoPy. (Mars 2020). Available : https ://docs.pycom.io/gitbook/assets/specsheets/Pycom-002-Specsheets-LoPy4-v2.pdf.

[91] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui, and T. Watteyne. "Understanding the Limits of LoRaWAN". In : IEEE Communication Magazine. Vol 55. pp 34–40. (2017).

[92] A. Laya, C. Kalalas, F. Vazquez-Gallego, L. Alonso, and J. Alonso-Zarate. "Goodbye, ALOHA !". In : IEEE Access. Vol 4. pp 2029–2044. (2016).

[93] T. Polonelli, D. Brunelli, and L. Benini. "Slotted aloha overlay on lorawan - a distributed synchronization approach". In : the 16th International Conference on Embedded and Ubiquitous Computing (EUC). pp 129-132. (2018).

[94] V. Giner, J. Martinez-Bauset, and C. Portillo. "Performance evaluation of framed slotted ALOHA with reservation packets and successive interference cancelation for M2M networks". In : Computer Networks Journal. Vol 155. pp 15-30. Doi : 10.1016/J.COMNET.2019.02.021. (2019).

[95] T. Polonelli, D. Brunelli, A. Marzocchi, and L. Benini. "Slotted aloha on lorawan design, analysis, and deployment. Sensors". Vol 19. Doi :10.3390/s19040838. (2019).

[96] J. Lee, W. Jeong, B. Choi. "A scheduling algorithm for improving scalability of LoRaWAN". In : International Conference on Information and Communication Technology Convergence (ICTC). pp 1383–1388. (2018).

[97] F. Moine. "Quel réseau pour l'internet des objets ?". Available : https ://fr.slideshare.net/Reseauxetservicestpa/rs-10-juin-2015-lora-franck. (2015).

[98] B. Reynders, Q. Wang, P. Tuset-Peiro, X. Vilajosana, and S. Pollin. "Improving reliability and scalability of LoRaWANs through lightweight scheduling". In : IEEE Internet Things Journal. Vol 5. pp 1830–1842. Doi : 10.1109/JIOT.2018.2815150. (2018).

[99] J. Haxhibeqiri, I. Moerman, J. Hoebeke. "Low overhead scheduling of LoRa transmissions for improved scalability". In : IEEE Internet of Things Journal. Vol 6. pp 3097–3109. Doi : 10.1109/JIOT.2018.2878942. (2018).

[100] F. Bonafini, A. Depari, P. Ferrari, A. Flammini, M. Pasetti, S. Rinaldi, E. Sisinni, M. Gidlund. "Exploiting localization systems for LoRaWAN transmission scheduling in industrial applications". In : the 15th IEEE International Workshop on Factory Communication Systems (WFCS). Sweden. pp 1–8. (2019).

[101] K.Q. Abdelfadeel, D. Zorbas, V. Cionca, and D. Pesch. "FREE–Fine-grained scheduling for reliable and energy efficient data collection in LoRaWAN". In : IEEE Internet of Things Journal. Vol 7. pp 669–683. Doi : 10.1109/JIOT.2019.2949918. (2020).

[102] D. Zorbas, K.Q. Abdelfadeel, V. Cionca, D. Pesch, and B. O'Flynn. "Offline scheduling algorithms for time-slotted LoRa-based bulk data transmission". In : IEEE 5th World Forum on Internet of Things (WFIoT). Limerick Ireland. pp 1–6. (2019).

[103] R. Piyare, A.L. Murphy, M. Magno, and L. Benini. "On-demand LoRa : Asynchronous TDMA for energy efficient and low latency communication in IoT". In : Sensors. Doi : 10.3390/s18113718. (2018).

[104] A. Lavric, and A.I. Petrariu. "LoRaWAN communication protocol : The new era of IoT. In : the International Conference on Development and Application Systems (DAS). Romania. pp 74-77. (2018).

[105] B. Bruhadeshwar, K. Kothapalli, and I.R. Pulla. "A fully dynamic and self-stabilizing TDMA scheme for wireless adhoc networks". In : the 14th IEEE International Conference on Advanced Information Networking and Applications. Australia. pp 511–518. (2010).

[106] P. Djukic, and S. Valaee. "Delay Aware Link Scheduling for Multi-Hop TDMA Wireless Networks". In : IEEE/ACM Transaction on Networking. Vol 17. pp 870–883. Doi :10.1145/1569732.1569747. (2009).

[107] T. Salonodis, and L. Tassiulas. "Asynchronous TDMA adhoc networks : scheduling and performance". In : European Transactions on Telecommunications. Vol 15. pp 391–403. Doi : 10.1002/ett.988. (2004).

[108] L. Labs. "Symphony Link a Revolutionary Wireless System for Wide-Area IoT Networks". Available : https ://www.link-labs.com/symphony. (2017).

[109] LoRa Alliance. "LoRaWAN Application Layer Clock Synchronization Specification v1.0.0". (2018).

[110] NIST Computer Security Division's (CSD), Security Technology Group (STG). "Block-cipher modes".

[111] C. De Cannière, A. Biryukov, and B. Preneel. "An introduction to Block Cipher Cryptanalysis". In : Proceedings of the IEEE. Vol 94. Doi : 10.1109/JPROC.2005.862300. (2006).

[112] X. Yang, E. Karampatzakis, C. Doerr, and F. Kuipers. "Security vulnerabilities in LoRaWAN". In the 13th International Conference on Internet of Things Design and Implementation (IoTDI). Orlando USA. pp 129–140. (2018).

[113] J. Lee, D. Hwang, J. Park, and K.H. Kim. "Risk analysis and countermeasure for bit-flipping attack in LoRaWAN". In : the International Conference on Information Networking (ICOIN). Da Nang Vietnam. pp. 549–551. (2017).

[114] L. Labs. "A comprehensive look at low power wide area networks for internet of things engineers and decision makers". p 16. (2016).

[115] R. Miller. "LoRa security - building a secure LoRa solution". Available : https : //labs.mwrinfosecurity.com, 2016. (2016).

[116] H. Lipmaa, P. Rogaway, and D. Wagner. (2000). "Ctr-mode encryption". In : the First NIST Workshop on Modes of Operation.

[117] Yang, Xueying. "LoRaWAN : Vulnerability Analysis and Practical Expoitation". TU Delft Electrical Engineering, Mathematics and Computer Science Thesis. (2017).

[118] F. L. Coman, K. M. Malarski, M. N. Petersen and S. Ruepp. "Security Issues in Internet of Things : Vulnerability Analysis of LoRaWAN, Sigfox and NB-IoT". In Global IoT Summit (GIoTS). Aarhus Denmark. pp 1-6. Doi : 10.1109/GIOTS.2019.8766430. (2019).

[119] E. Aras, G.S. Ramachandran, P. Lawrence, and D. Hughes. "Exploring the security vulnerabilities of LoRa". In the 3rd IEEE International Conference on Cybernetics (CYBCONF). Exeter UK. (2017).

[120] S. Zulian. "Security threat analysis and countermeasures for LoRaWAN join procedure". http ://tesi.cab.unipd.it/53210/. (2016).

[121] S. Tomasin, S. Zulian, and L. Vangelista. "Security analysis of LoRaWAN join procedure for internet of things networks". In : Wireless Communications and Networking Conference Workshops (WCNCW). San Francisco USA. pp 1–6. (2017).

[122] S. Na, D. Hwang, W. Shin, and K.-H. Kim. "Scenario and countermeasure for replay attack using join request messages in LoRaWAN". In : International Conference on Information Networking (ICOIN). Da Nang Vietnam. pp 718–720. (2017).

[123] J. Michorius. "What's Mine is Not Yours : LoRa network and privacy of data on publishing devices". In : the 25th Twente Student Conference on IT. (2016).

[124] S. Naoui, M. E. Elhdhili, and L. A. Saidane. "Enhancing the security of the IoT LoRaWAN architecture". In : International Conference on Performance Evaluation and Modeling in Wired and Wireless Networks (PEMWN). Paris France. pp. 1–7. (2016).

[125] E. Aras, N. Small, G. S. Ramachandran, S. Delbruel, W. Joosen, and D. Hughes. "Selective jamming of LoRaWAN using commodity hardware". Doi : 10.1145/3144457.3144478. (2017).

[126] L.V Houtven. "Crypto 10"1. The Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0). p. 254.

[127] W. Stallings. "The offset codebook (OCB) block cipher mode of operation for authenticated encryption". In : Cryptologia Journal. Vol 42. pp 135-145. Doi : 10.1080/01611194.2017.1422048. (2018).

[128] B. Prashanth Namatheertham, R. Suppala, B. Govindool, and P. Ravi Kiran. "Software implementation of OCB mode". (2006).

[129] P. Rogaway, M. Bellare, and J. Black. "OCB : A Block-Cipher Mode of Operation for Efficient Authenticated Encryption". In : ACM Journal Name. Vol 5. p 39. (2003).

[130] E. Anceaume, R. Ludinard, B. Sericola, F. Tronel. "Modélisation et Évaluation des Attaques Ciblées dans un Overlay Structuré". In : Colloque Francophone sur l'Ingénierie des Protocoles. Sainte Maxime, France. (2011).

[131] P. Błaśkiewicz, J. Cichoń, M. Kutyłowski, M. Zawada. "Revised Gateway Selection for LoRa Radio Networks". In : Ad-Hoc, Mobile, and Wireless Networks (ADHOC-NOW). Vol 11803. Springer, Cham. https ://doi.org/10.1007/978-3-030-31831-4-16. (2019).

[132] A. A. Tesfay, E. P. Simon, I. Nevat and L. Clavier. "Multiuser Detection for Downlink Communication in LoRa- Like Networks". In : IEEE Access. Vol 8. pp 199001-199015. Doi : 10.1109/ACCESS.2020.3034973. (2020).

[133] S. Abboud, N. El Rachkidy, A. Guitton, H. Safa. "Gateway Selection for Downlink Communication in LoRaWAN". In : IEEE Wireless Communications and Networking Conference. Marrakech Morocco. (2019).